



**MINI PROJECT REPORT  
ON  
CAMPUS TRACKER  
SUBMITTED BY**

- 1) Tejal Patil**
- 2) Hrishikesh Mali**
- 3) Harshada Deore**
- 4) Ajay Badale**
- 5) Shubham Gupta**

**Under the Guidance of  
Prof. Milind Deshkar**

**Submitted To**

**Savitribai Phule Pune University**

**As a partial fulfillment for the award of the degree  
of**

**MASTER OF COMPUTER APPLICATION**

**(MCA) Semester: 1**

**At  
ASM'S**

**Institute of Business Management and Research,  
Chinchwad, Pune – 19**

**(Affiliated to SPPU & Approved by AICTE)**

**Session: 2021-23**

## INDEX

<b>Sl. No</b>	<b>Particulars</b>	<b>Page No.</b>
<b>1.</b>	<b>Introduction/Aims and Objective</b>	<b>01 - 02</b>
<b>2.</b>	<b>System Analysis:</b> <b>Identification of Need</b> <b>Scope of the Project</b>	<b>02 - 03</b>
<b>3.</b>	<b>Feasibility Study:</b> <b>Technical Feasibility</b> <b>Economic Feasibility</b> <b>Operational Feasibility</b>	<b>04 - 05</b>
<b>4.</b>	<b>Diagrams:</b> <b>ERD</b> <b>DFD</b> <b>USECASE DIAGRAMS</b> <b>SEQUENCE DIAGRAMS</b>	<b>05 - 10</b>
<b>5.</b>	<b>S/W &amp; H/W Requirement Specification</b>	<b>11 - 12</b>
<b>6.</b>	<b>Data Dictionary</b>	<b>12 - 17</b>
<b>7.</b>	<b>Screenshots</b>	<b>18 - 62</b>
<b>8.</b>	<b>Validation Checks</b>	<b>63 - 66</b>
<b>9.</b>	<b>Sample Code</b>	<b>67- 82</b>
<b>10.</b>	<b>Implementation and Maintenance</b>	<b>83 - 95</b>
<b>11.</b>	<b>Future scope of the Mini Project</b>	<b>96 -97</b>
<b>12.</b>	<b>Bibliography/References/Glossary</b>	<b>97</b>

### **1. Introduction/Aims and Objective:**

The main objective of the Campus Tracker is to manage the details of College, Company, Job, Students. The project is totally built at the administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the College, Company, Placement, Job. It tracks all the details about the Company, Job, students.

A college campus tracker that consists of a student login, Company login, placement officer and admin login . The project is beneficial for college students, various companies visiting the campus for recruitment, and even the college placement officer. The software system allows the students to create their profiles and upload all their details including their marks onto the system. The admin can check each student's details and can remove faulty accounts.

The system also consists of a company login where various companies visiting the college can view a list of students in that college and also their respective resumes. The software system allows students to view a list of companies who have posted for vacancies. The admin has overall rights over the system and can moderate and delete any details not pertaining to college placement rules. The system handles student as well as company data and efficiently displays all this data to respective sides.

### **2. System Analysis:**

- Identification of Need:**

#### **Existing System:**

Companies that came to campus visit for recruitment purposes will have specific qualification criteria. Qualification criteria include marks or grade range, specific program. Currently, companies coming for the recruitment used to communicate with one college or institution and ask for the details of eligible students. Then a college or institution tries to spread the same to peer institutions and pool up the data. Then filters the number of students and after this, they used to send feedback to a company. This involves much of the time and this time has to be saved.

#### **Problems in Existing System:**

- We need to maintain all the student record manually. We won't be able to track the student details.
- This system has more paperwork. Student are only dependent on the placement officer.
- The current system might have loss of data.

### **Proposed System:**

It is proposed to conduct the process of recruitment of suitable candidates. aims to compensate the deserving candidates comparable with the best in the industry, it is determinedly focused not to compromise on the quality of manpower intake since the selected ones would have a long road of achievements ahead well supported by Requirement from the Institute We solicit your active assistance in bringing this rare opportunity to your present as well as the past students for their benefit by mailing a copy of these documents to them in order to enable them to forward their Colleges/Students to us. Alternatively, if the list of present and past students is made available to us, we shall be making arrangements to do the same needful directly. Additionally, we would request you to kindly display a copy of these documents at your Main Notice Board so that a maximum number of students are able to avail the benefit of this extremely attractive offer.

### **Advantages of the proposed system:**

- There will be no need of maintaining notice or emailing each and every student about the company which is visiting the college.
- The students can keep updated themselves through this software.
- The company can view all students' detail and the system can shortlist students according to their criteria instead of doing it manually.
- There is admin login that can view and manage students' documents and company's account. Students can register online instead of going to placement department for registration.
- This system saves time and efforts. Reduce paper works.
- The system makes the overall project management much easier and flexible.

### **• Scope of the Project :**

It may help collecting perfect recruitment details. In a very short time, the collection will be obvious, simple and sensible. It will help a person to know the management of passed year perfectly. It also helps in current all works relative to Campus Tracker. It will also reduce the cost of collecting the management & collection procedure will go on smoothly.

### **3. Feasibility Study:**

Feasibility study is conducted once the problem is clearly understood. Feasibility is a high level capsule version of the entire system analysis and design process. The objective is to determine quickly at the minimum expense how to solve the problem. The purpose of feasibility is not to solve the problem but to determine if the problem is worth solving.

The system has been tested for feasibility in the following points.

1. Technical Feasibility
2. Economical Feasibility
3. Operational Feasibility

- **Technical Feasibility :**

The project entitled "Campus Tracker" is technically feasible because of the below mentioned feature. The project was developed in JAVA which graphical user interface. It provides high level of reliability, availability and compatibility. All these make Java an appropriate Language for this project. Thus the existing software Java is a powerful language.

- **Economic Feasibility :**

The computerized system will help in automate the selection leading the profits and details of the organization. With this software, the machine and man power utilization are expected to go up by 80-90% approximately. The costs incurred of not creating a system are set to be great, because precious time can be wasted manually.

- **Operational Feasibility :**

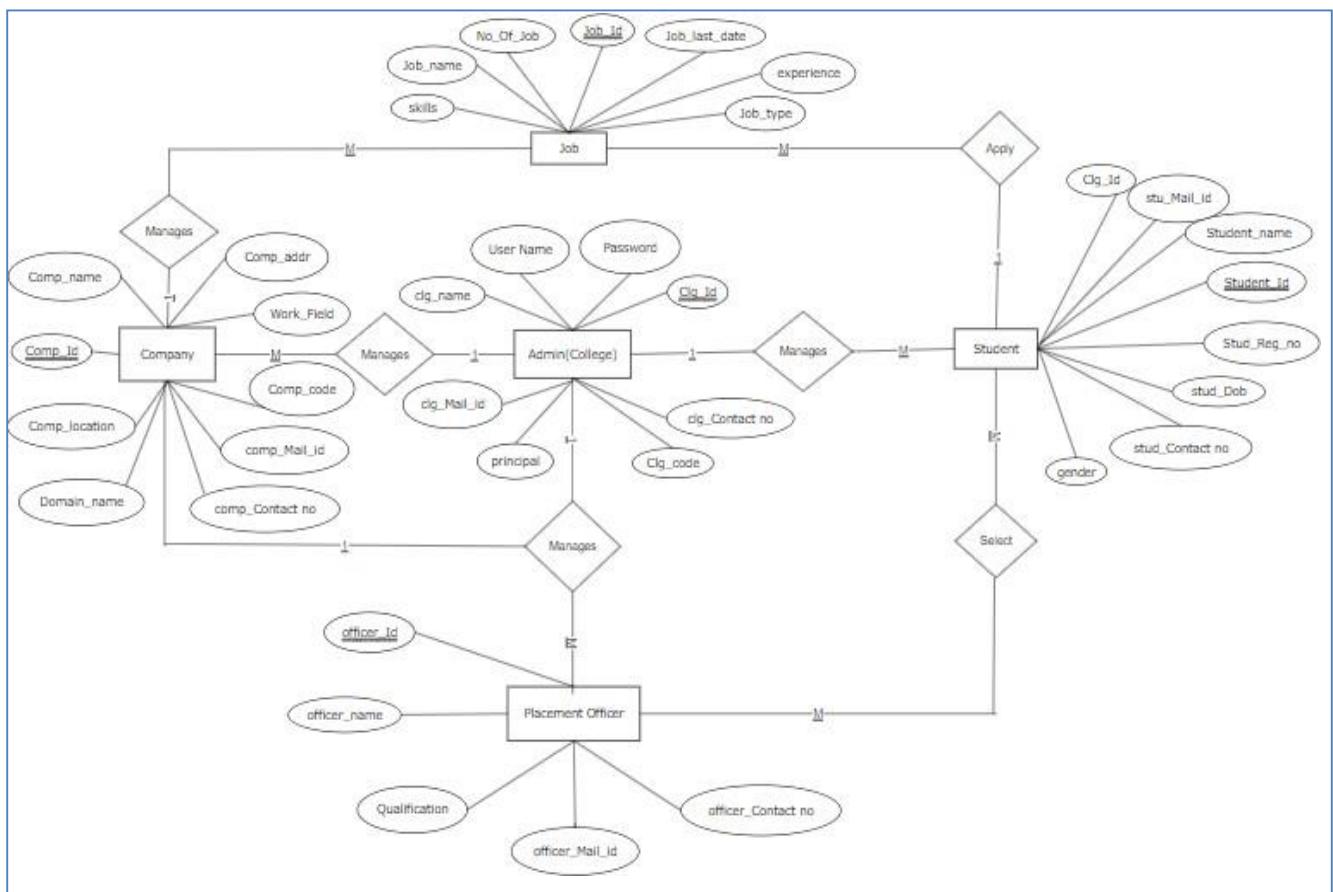
In this project, the management will know the details of each project where he may be presented and the data will be maintained as decentralized and if any inquiries for that particular contract can be known as per their requirements and necessities. Time evaluation is the most important consideration in the development of project. The time schedule required for the development of this project is very important since more development time affects machine time, cost and cause delay in the development of other systems. A reliable Campus Tracker can be developed in the considerable amount of time.

## CAMPUS TRACKER

---

### 4. Diagrams:

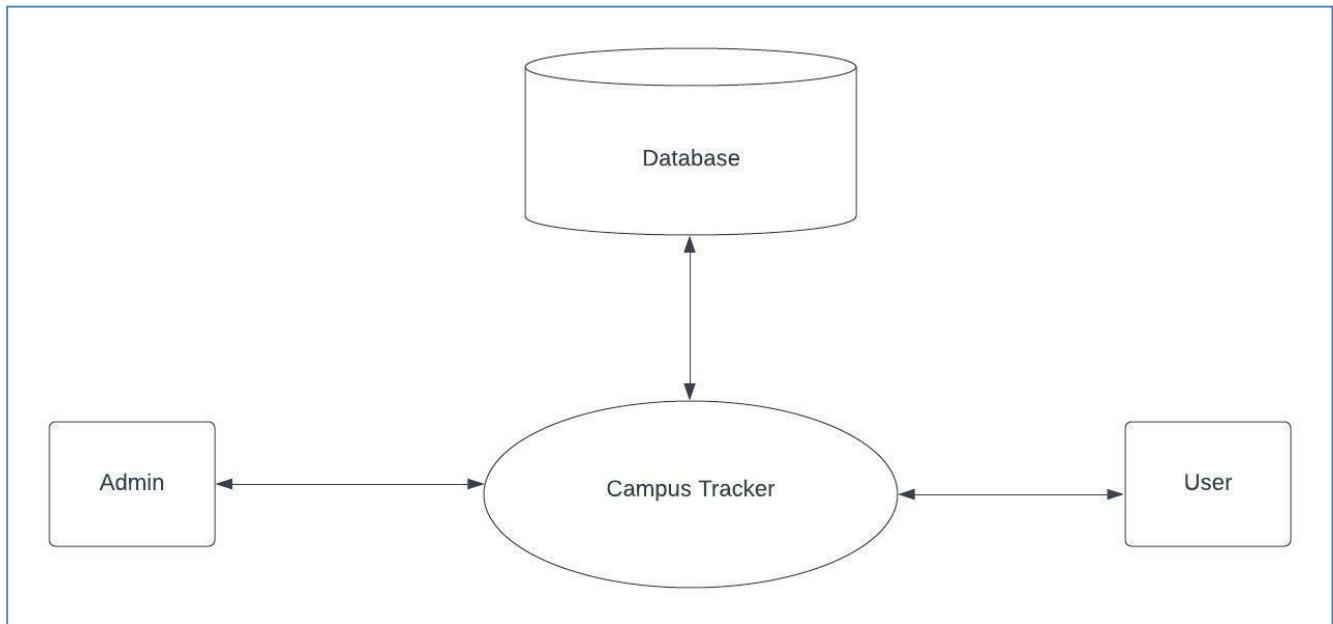
#### ERD DIAGRAM



# CAMPUS TRACKER

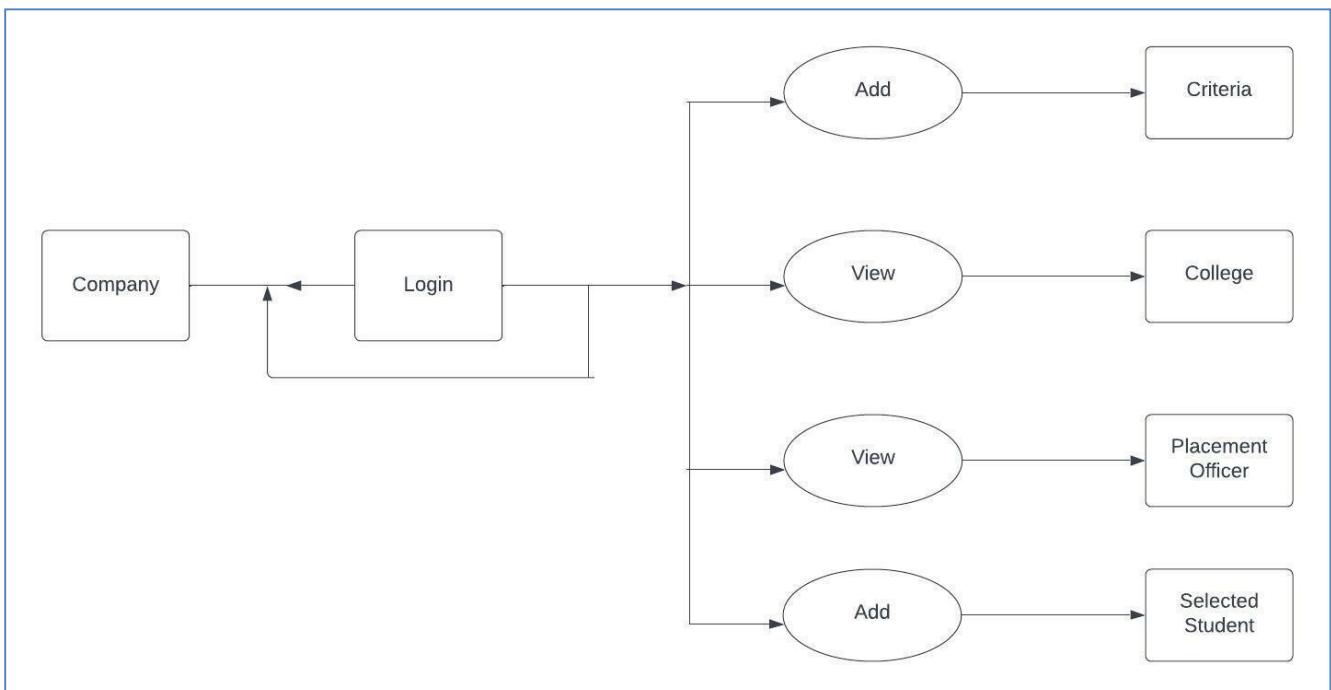
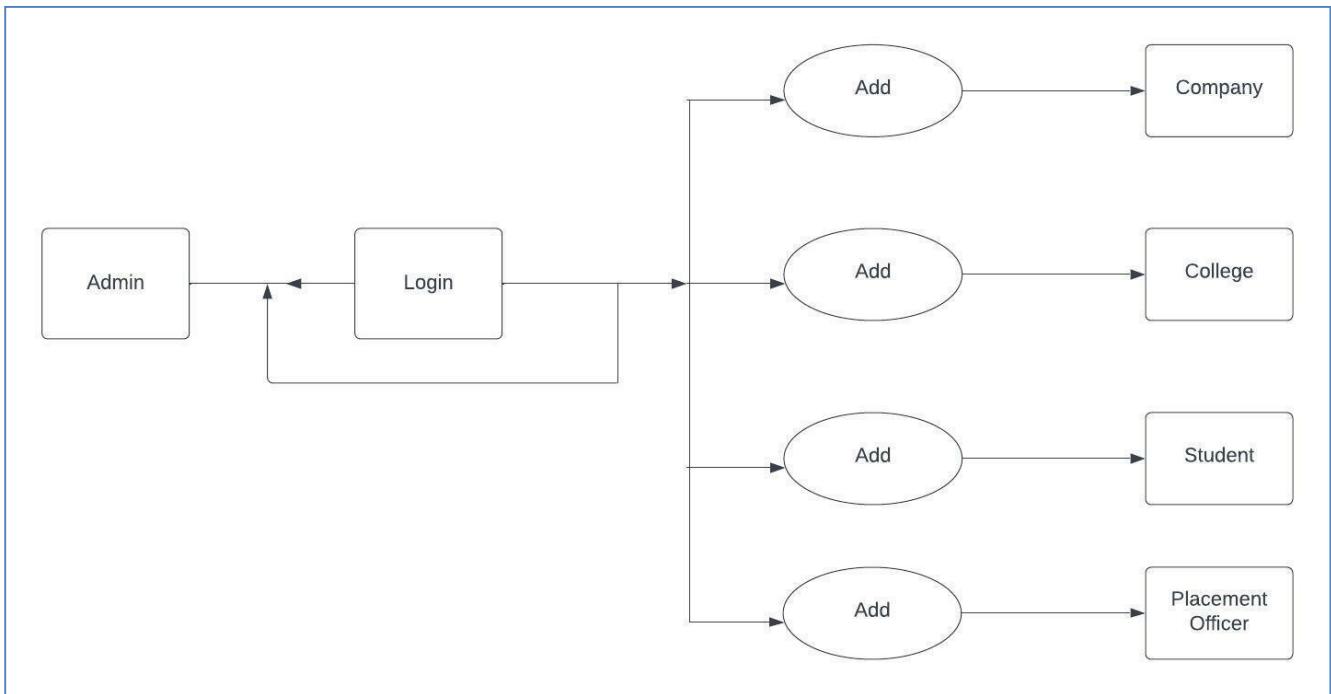
## DATA FLOW DIAGRAM

**0 Level Diagram:**

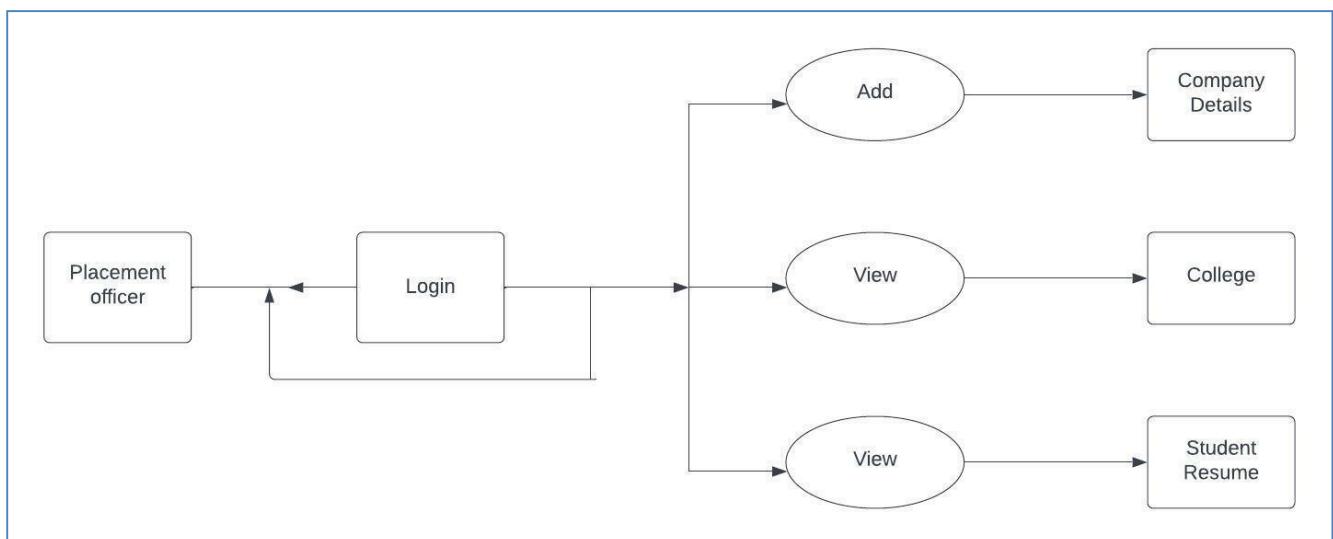
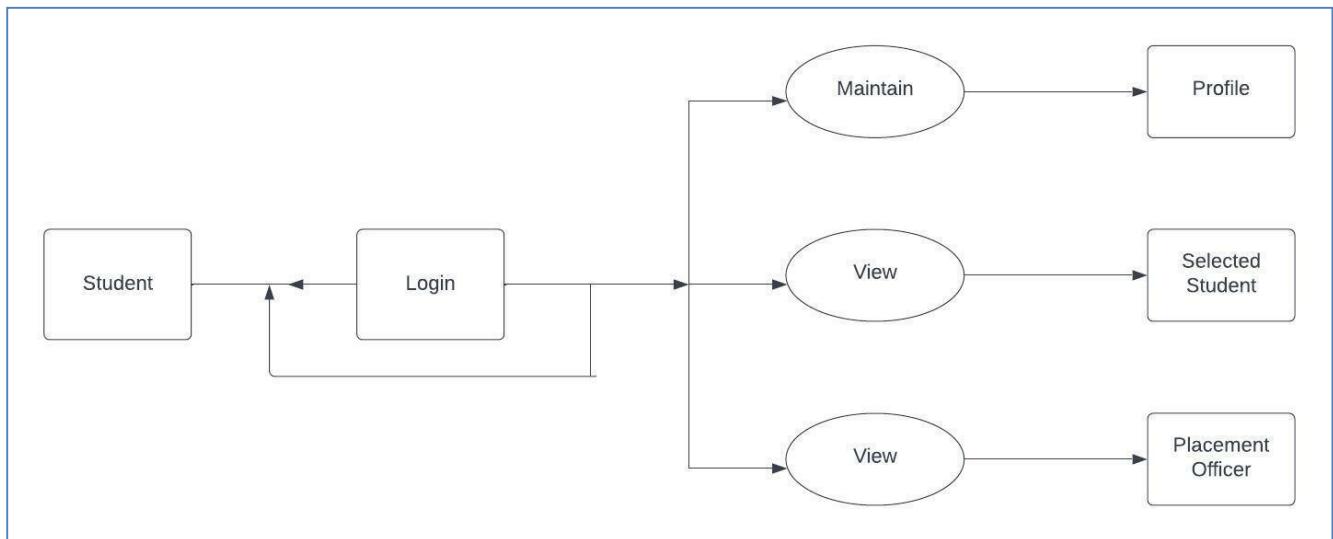


# CAMPUS TRACKER

## 1 st Level Diagram:

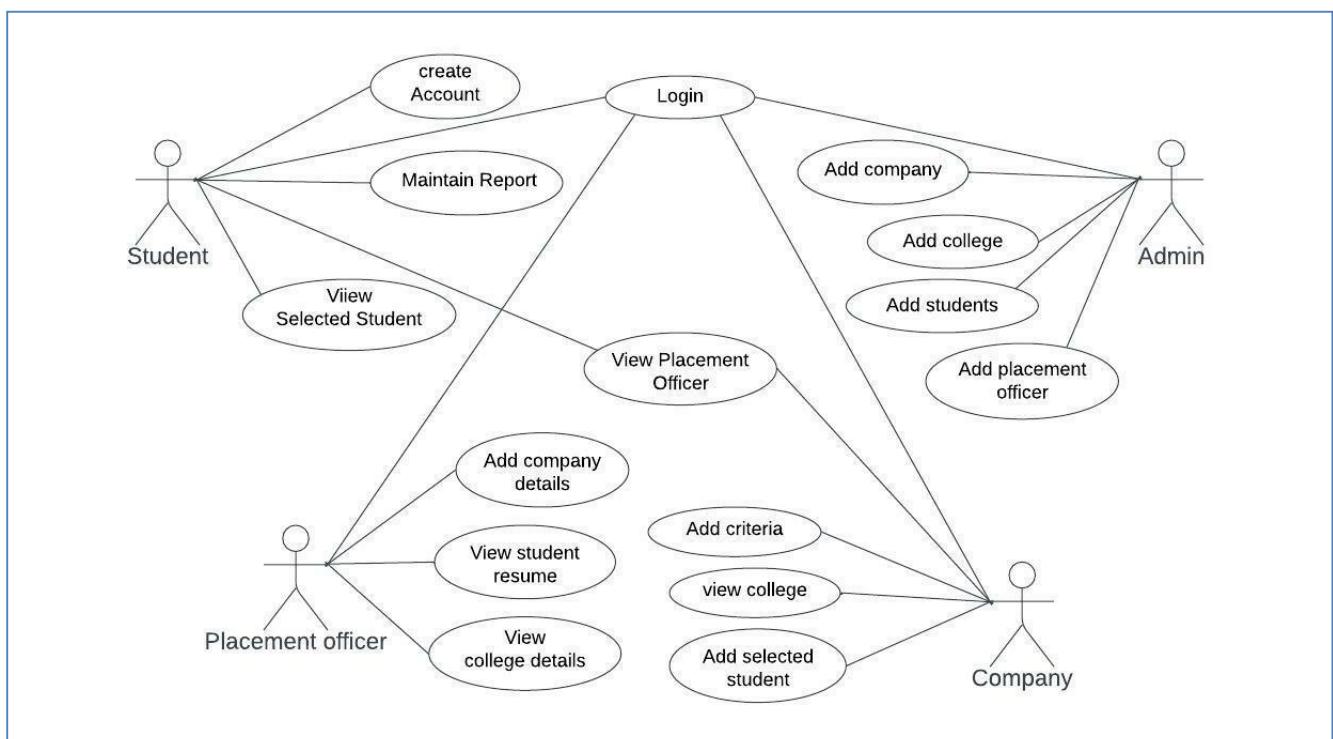


## CAMPUS TRACKER



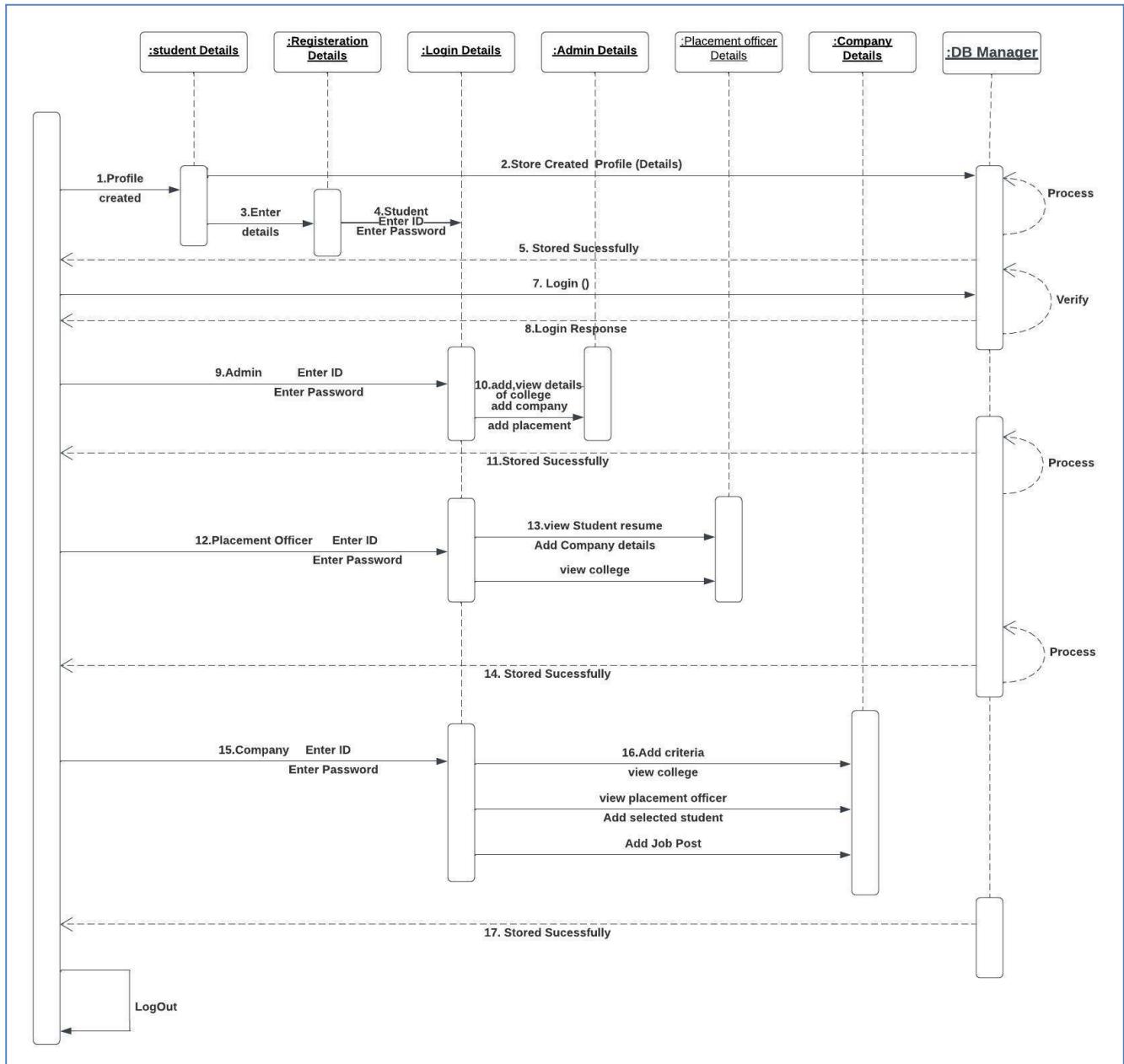
## CAMPUS TRACKER

## USE CASE DIAGRAM



# CAMPUS TRACKER

## SEQUENCE DIAGRAM



## 5. S/W & H/W Requirement Specification:

### H/W Configuration (Server Side):

- Processor – Core i3 & above
- Speed - 2 GHz & above
- RAM – 2 GB & above

### Software (Server Side):

Operating System: Windows 10,11

Front end: CSS style sheet/HTML tags

Middle ware: Java Development Kit(JDK 1.5.0)

Backend: MySQL Database Server

Server: Apache Tomcat, Wamp

### H/W Configuration (Client Side):

- Processor –Core i3 & above
- Speed – 2 GHz & above
- RAM – 2 GB & above

### Software (Client Side):

Operating System: Windows 10,11

Web Browser: Google Chrome, Mozilla Firefox,etc.

## CAMPUS TRACKER

---

### 6. Data Dictionary:

**Cardinality** - In SQL (Structured Query Language), the term cardinality refers to **the uniqueness of data values contained in a particular column (attribute) of a database table.**

#### campus details:

Field	Type	Null	Default	Comments
campus_id	int(11)	No		
campus_name	varchar(50)	No		
placement_officer_id	int(11)	No		
company_name	varchar(50)	No		
skills	varchar(50)	No		
job_type	varchar(50)	No		
experience	varchar(50)	No		
campus_state	varchar(50)	No		
campus_timing	datetime	No		

Keyname	Type	Cardinality	Field
<b>PRIMARY</b>	PRIMARY	0	campus_id

#### college details:

Field	Type	Null	Default	Comments
college_id	int(11)	No		
college_name	varchar(50)	No		
principal	varchar(50)	No		
email_id	varchar(35)	No		
contact_no	varchar(50)	No		
college_code	int(11)	No		

## CAMPUS TRACKER

---

Keyname	Type	Cardinality	Field
PRIMARY	PRIMARY	3	college_id

### **company details :**

Field	Type	Null	Default	Comments
company_id	int(11)	No		
company_name	varchar(50)	No		
company_address	varchar(200)	No		
domain_name	varchar(50)	No		
work_field	varchar(50)	No		
company_email_id	varchar(50)	No		
company_contact_no	varchar(13)	No		
company_code	varchar(11)	No		

Keyname	Type	Cardinality	Field
PRIMARY	PRIMARY	1	company_id

### **Criteria:**

Field	Type	Null	Default	Comments
id	int(100)	No		
Name	varchar(100)	No		
sslc	varchar(100)	No		
puc	varchar(100)	No		
degree	varchar(100)	No		
masters	varchar(100)	No		

Keyname	Type	Cardinality	Field
PRIMARY	PRIMARY	1	id

## CAMPUS TRACKER

---

### **job\_details:**

Field	Type	Null	Default	Comments
job_details_id	int(11)	No		
job_name	varchar(50)	No		
company_name	varchar(50)	No		
number_of_jobs	int(11)	No		
job_type	varchar(50)	No		
skills	varchar(50)	No		
experience	varchar(50)	No		
job_lat_date	date	No		

Keyname	Type	Cardinality	Field
<b>PRIMARY</b>	PRIMARY	2	job_details_id

### **job\_post :**

Field	Type	Null	Default	Comments
job_post_id	int(11)	No		
job_name	varchar(50)	No		
company_name	varchar(50)	No		
number_of_employees	int(11)	No		
job_date	date	No		
experience	varchar(50)	No		
skills	varchar(50)	No		
job_type	varchar(50)	No		

Keyname	Type	Cardinality	Field
<b>PRIMARY</b>	PRIMARY	0	job_post_id

## CAMPUS TRACKER

---

### Login:

Field	Type	Null	Default	Comments
User_name	varchar(50)	No		
Password	varchar(50)	No		
Type	varchar(50)	No		
Hint_Q	varchar(50)	No		
Hint_A	varchar(50)	No		
Status	varchar(50)	No		

### Marks:

Field	Type	Null	Default	Comments
student_no	int(11)	No		
reg_no	varchar(100)	No		
student_name	varchar(100)	No		
sslc_marks	varchar(10)	No		
puc_marks	varchar(10)	No		
degree_marks	varchar(10)	No		
masters	varchar(11)	No		

Keyname	Type	Cardinality	Field
<b>PRIMARY</b>	PRIMARY	7	student_no

### placement officer:

Field	Type	Null	Default	Comments
placement_office_id	int(11)	No		
placement_officer_name	varchar(50)	No		
qualification	varchar(50)	No		
email_id	varchar(50)	No		
contact_no	varchar(50)	No		

## CAMPUS TRACKER

---

Keyname	Type	Cardinality	Field
<b>PRIMARY</b>	PRIMARY	2	placement_office_id

### **Resume:**

Field	Type	Null	Default	Comments
cv_id	int(11)	No		
cv_name	varchar(50)	No		
discription	varchar(50)	No		

Keyname	Type	Cardinality	Field
<b>PRIMARY</b>	PRIMARY	0	cv_id

### **selected students:**

Field	Type	Null	Default	Comments
selected_student_id	int(11)	No		
student_reg	varchar(100)	No		
placement_officer_id	int(11)	No		
job_details_id	int(11)	No		
selected_date	varchar(12)	No		

Keyname	Type	Cardinality	Field
<b>PRIMARY</b>	PRIMARY	3	selected_student_id

## CAMPUS TRACKER

---

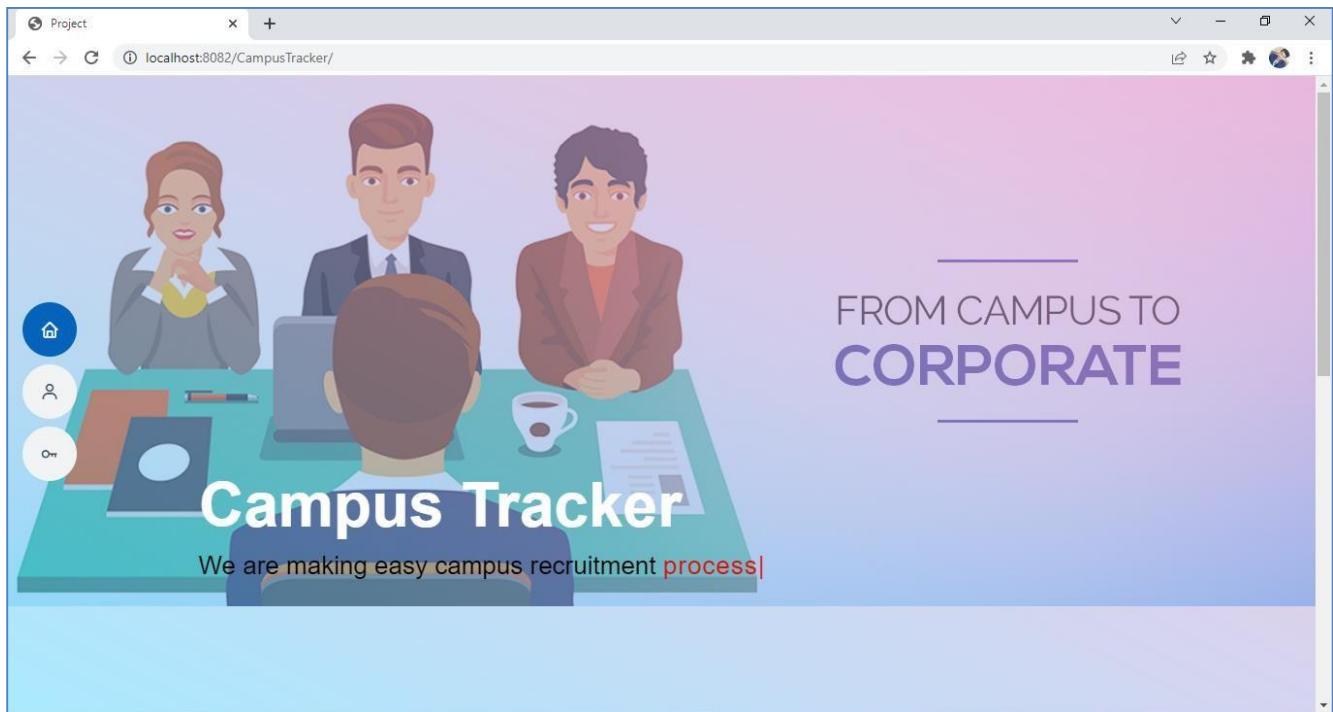
### **student details:**

Field	Type	Null	Default	Comments
stud_id	int(11)	No		
stud_name	varchar(50)	No		
stud_fathername	varchar(50)	No		
stud_dob	varchar(100)	No		
stud_gender	varchar(50)	No		
stud_email_id	varchar(25)	No		
stud_contact_no	varchar(50)	No		
stud_register_no	varchar(50)	No		
resume_upload	varchar(150)	No		
college_id	int(11)	No		

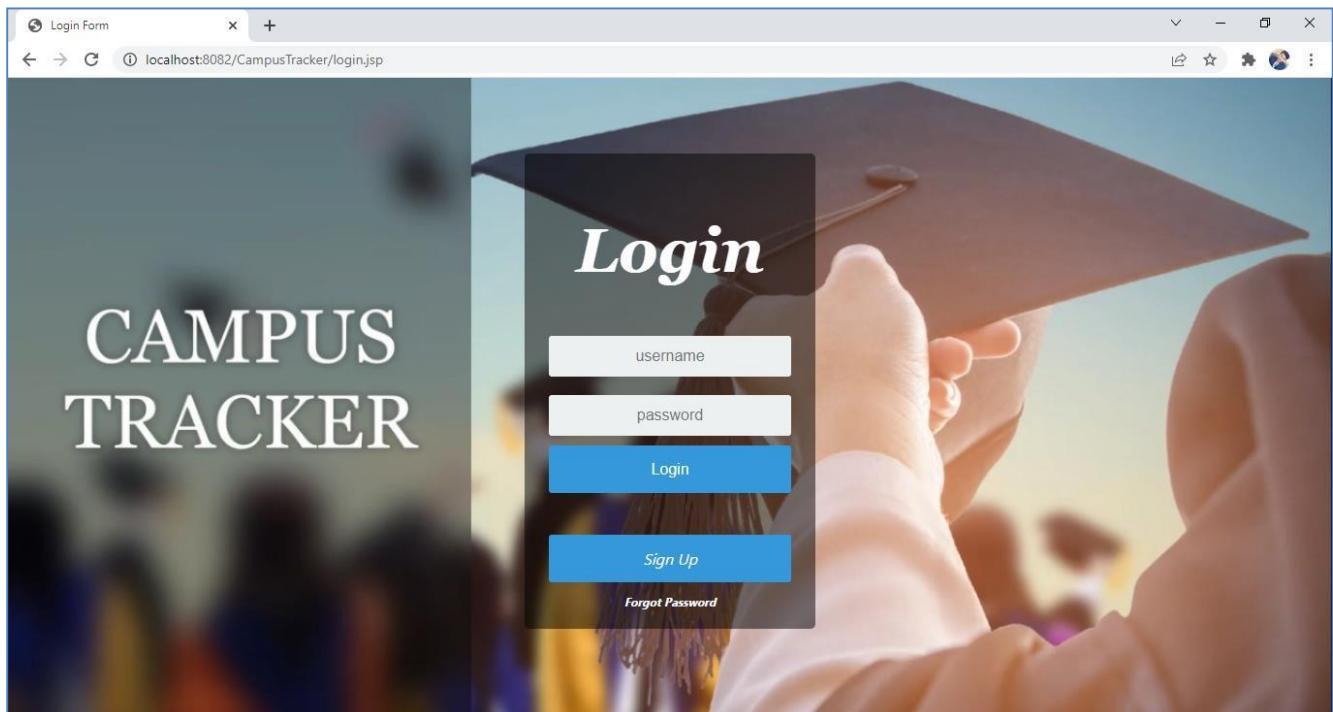
Keyname	Type	Cardinality	Field
<b>PRIMARY</b>	PRIMARY	8	stud_id

## CAMPUS TRACKER

### 7. Screenshots: Home Page

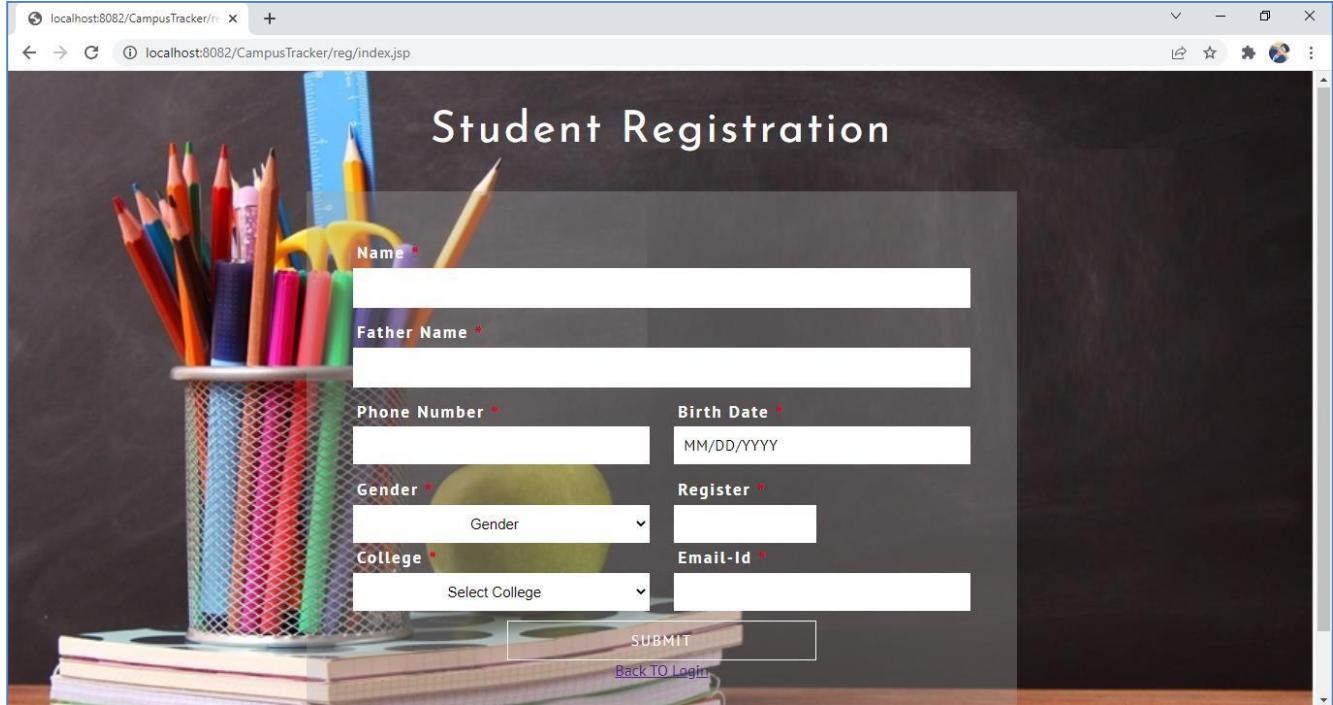


### Login Page:



## CAMPUS TRACKER

### Registration/ Sign Up Page: For Student



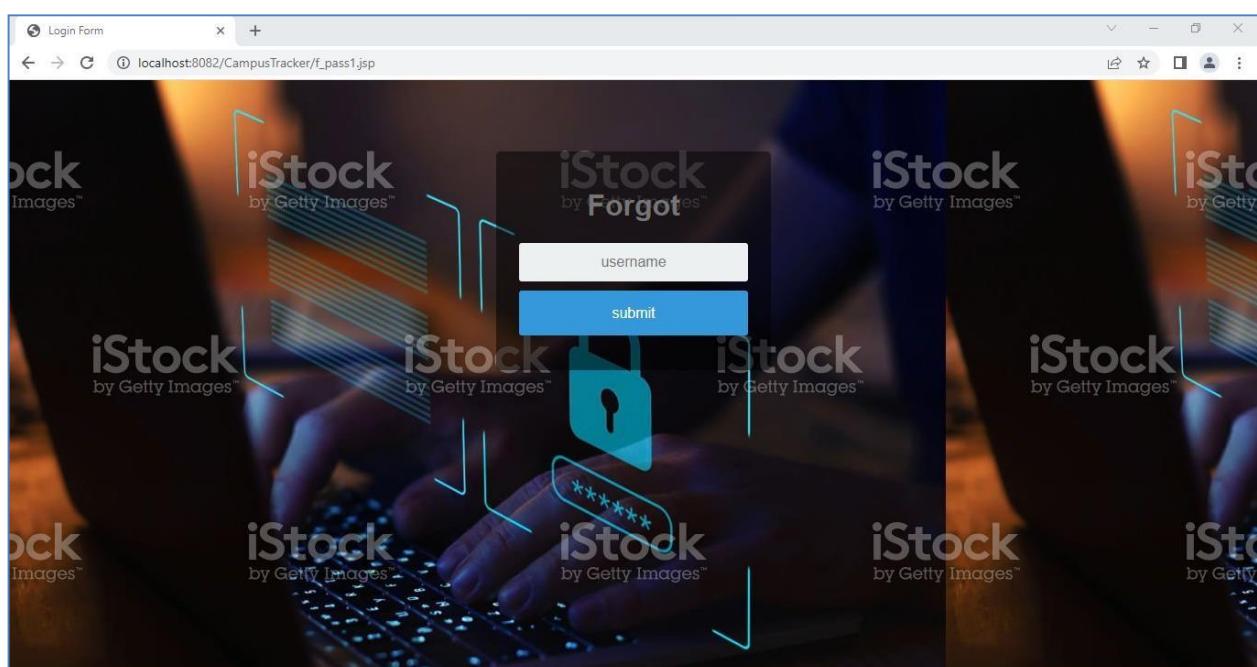
A screenshot of a web browser showing a "Student Registration" form. The background features a chalkboard with a pencil holder containing various colored pencils and a stack of books at the bottom. The form fields include:

- Name \*
- Father Name \*
- Phone Number \*
- Birth Date \*
- Gender \*
- Register \*
- College \*
- Email-Id \*

Buttons for "SUBMIT" and "Back TO Login".

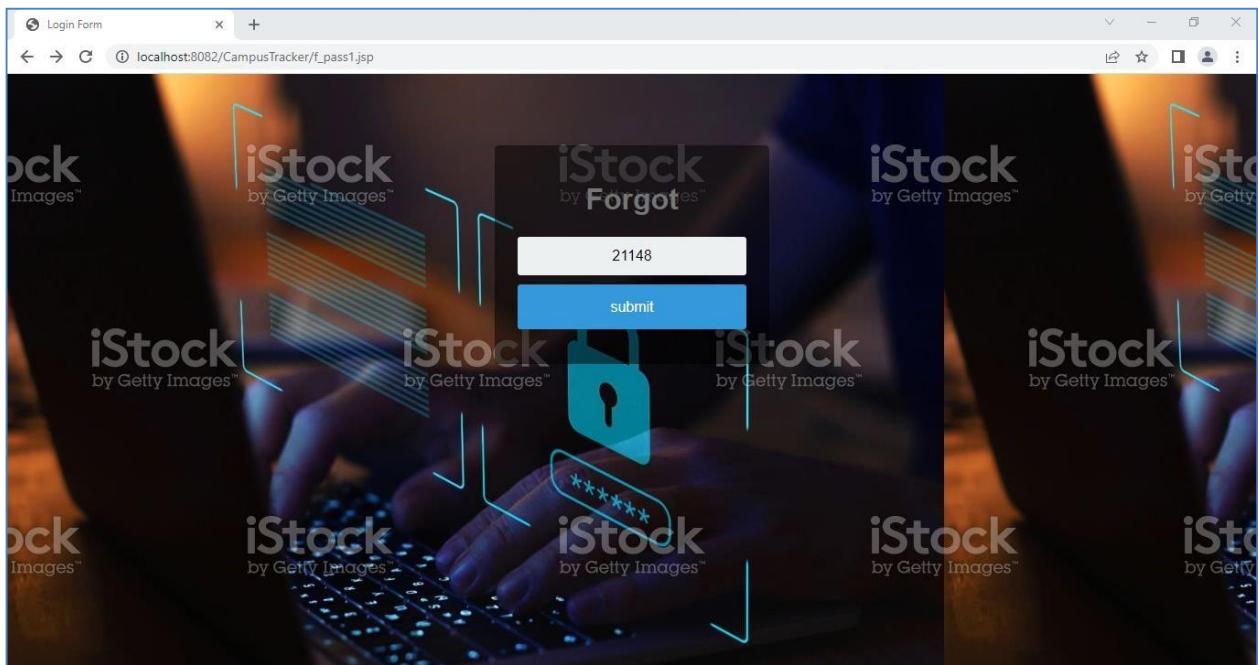
URL: localhost:8082/CampusTracker/reg/index.jsp

### Forgot Password:

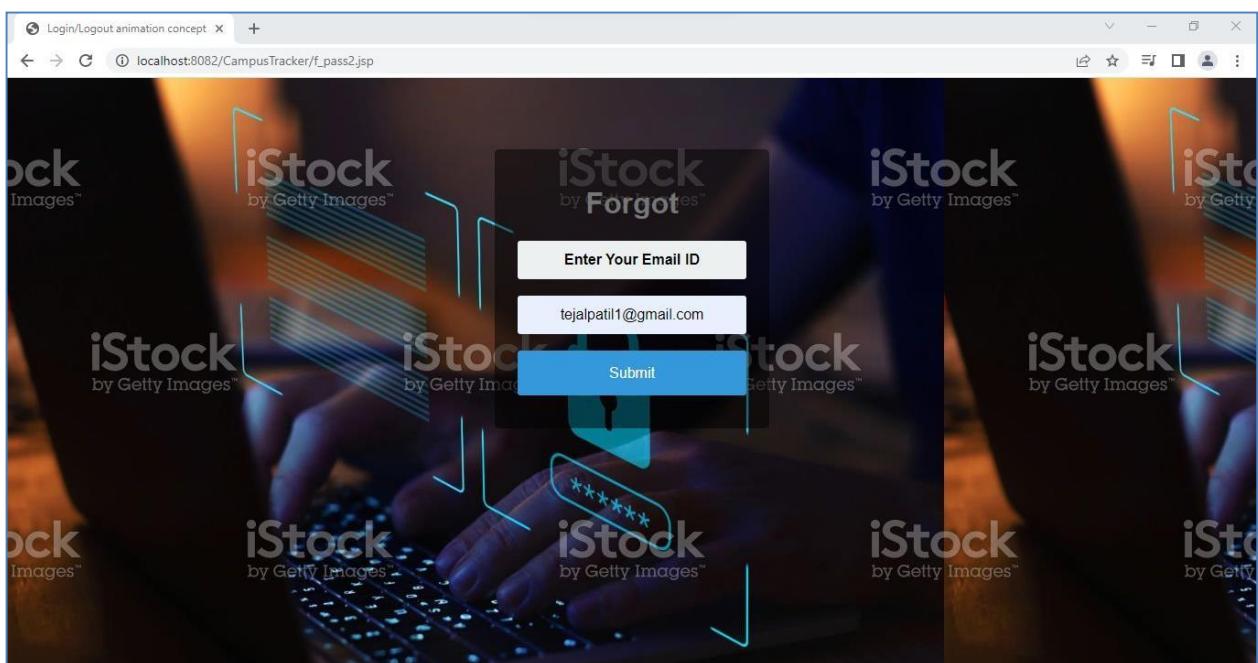


## CAMPUS TRACKER

Example: student login - 21148

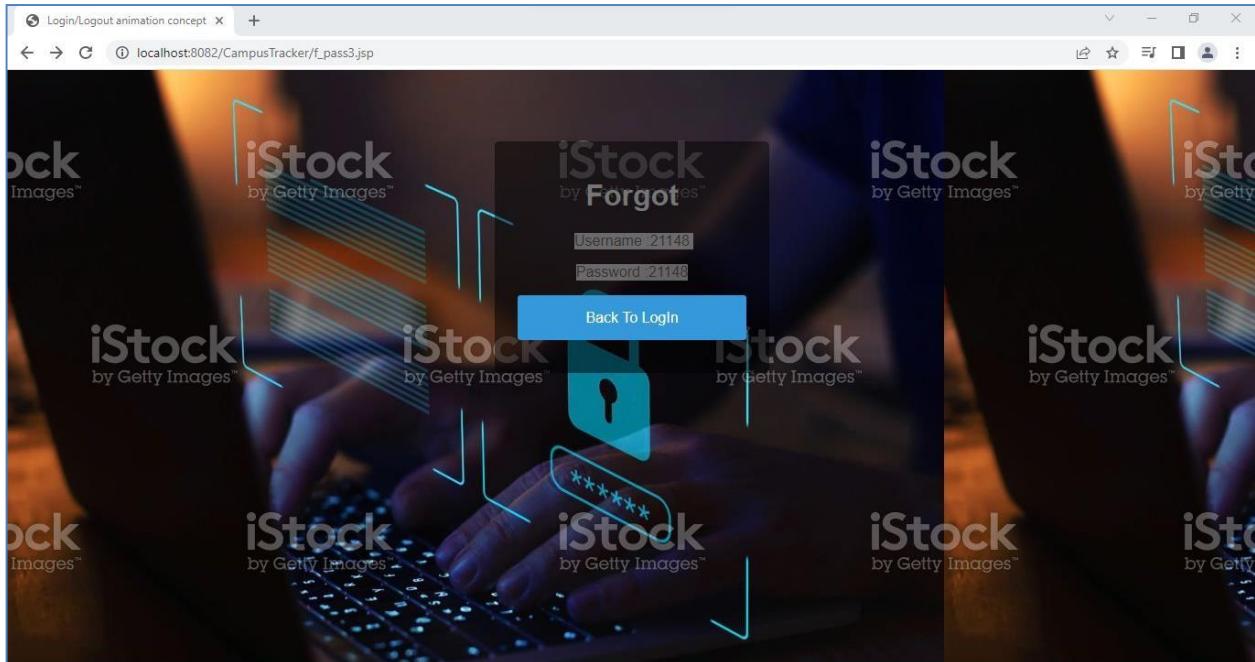


Enter Email Id: Student Mail Id - [tejalpatil1@gmail.com](mailto:tejalpatil1@gmail.com)



## CAMPUS TRACKER

Here, We get Username-21148 , Password-21148



Student Login:

For example: Username-21148 , Password-21148



# CAMPUS TRACKER

## Student Dashboard/Profile :

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/user/student\_details\_view.jsp". The page has a blue header with the "Campus Tracker" logo and a navigation bar for "Student". On the left is a sidebar menu with options like "Home", "My Profile", "Resume", etc. The main content area is titled "Student Details" and displays a table with one record:

Student id	Student name	Student full name	Student DOB	Student gender	Student email id	Student contact no	Student reg no	college_name	Edit
2	Tejal Patil	Rajaram Patil	2000/11/25	Female	tejalpatil1@gmail.com	9657910300	21148	ASM IBMR Pune	<a href="#">Edit</a>

Below the table, it says "Showing 1 to 1 of 1 entries". At the bottom, there are links for "Previous", "1", and "Next".

## Student login : Resume

### Function : Resume - Upload resume

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/user/resume\_view.jsp". The page has a blue header with the "Campus Tracker" logo and a navigation bar for "Student". On the left is a sidebar menu with options like "Home", "My Profile", "Resume", etc. The main content area is titled "Resume" and displays a table with one record:

Student id	Student Name	Reg No	Resume	Delete
2	Tejal Patil	21148	<a href="#">Download</a>	<a href="#">Remove</a>

Below the table, it says "Showing 1 to 1 of 1 entries". At the bottom, there are links for "Previous", "1", and "Next".

# CAMPUS TRACKER

## Function : Resume -Upload resume – Choose File

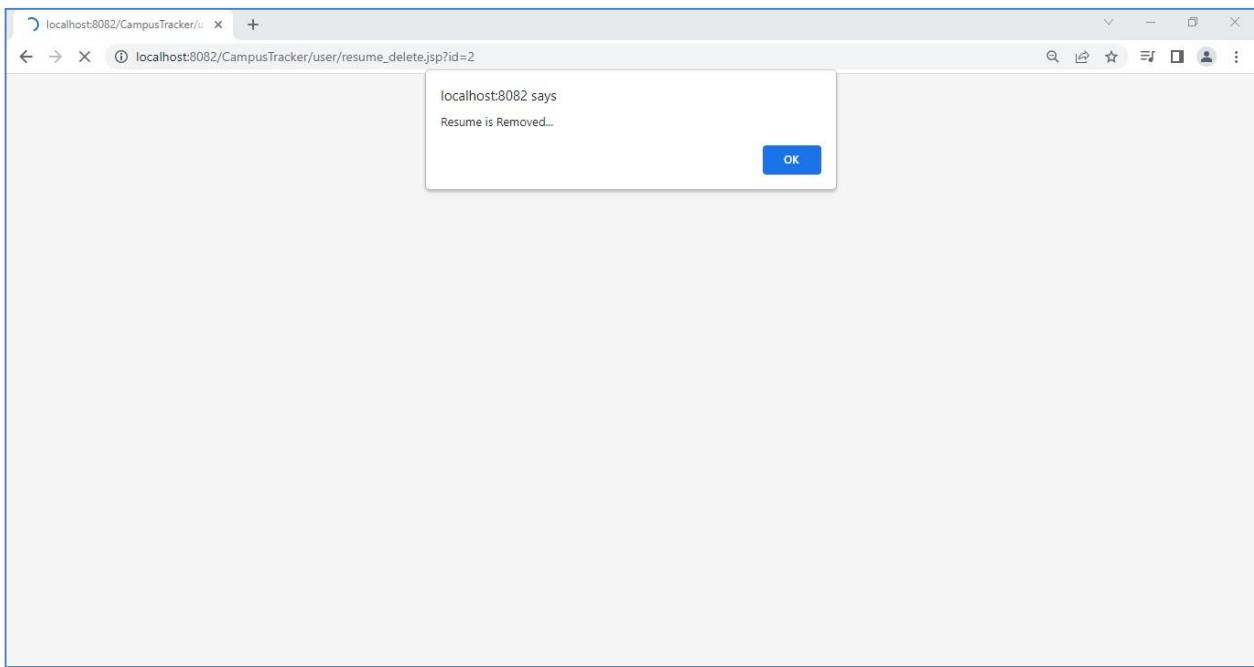
The screenshot shows a web browser window titled "Admin Home Page" at "localhost:8082/CampusTracker/user/resume.jsp". The page has a blue header with the title "Campus Tracker" and a group of people icon. A sidebar on the left contains a "Home" button and links to "My Profile", "Resume", "Selected students", "Job details", "college details", "Student Marks", "placement officer", "Selected in criteria", and "Change Password". The main content area is titled "Resume" and contains a form with an "Upload Resume" button, a "Choose file" input field containing "tejal\_patil\_CV.pdf", and "Submit" and "Reset" buttons. At the bottom left, it says "Developed by -Group Number- 27".

**Uploaded Successfully Resume**

The screenshot shows a web browser window at "localhost:8082/CampusTracker/user/resume\_insert.jsp". A modal dialog box is open, displaying the message "localhost:8082 says uploaded successfully" with an "OK" button.

# CAMPUS TRACKER

## Removed Resume



## Selected Students:

A screenshot of a web-based application titled "Admin Home Page" from "localhost:8082/CampusTracker/user/selected\_students\_view.jsp". The page features a header with the "Campus Tracker" logo and a navigation menu on the left. The main content area is titled "Student Details" and shows a table with one row of data. The table columns are: Selected student id, Register no., Placement officer Name, Job Name, Company Name, Selected date, and Delete. The data in the table is:

Selected student id	Register no.	Placement officer Name	Job Name	Company Name	Selected date	Delete
1	21148	Prof Milind	TCS Java Developer	TCS	2022-02-15	<a href="#">Delete</a>

At the bottom of the page, there is a footer with the text "Developed by -Group Number- 27".

# CAMPUS TRACKER

## Job Details :

The screenshot shows a web browser window for 'Campus Tracker' at localhost:8082. The title bar says 'Admin Home Page'. The main content area has a blue header 'Campus Tracker' with a group of people icon. On the left is a sidebar with links: Home, My Profile, Resume, Selected students, Job details, college details, Student Marks, placement officer, Selected in criteria, and Change Password. The main content area is titled 'Job Details' with a table showing one record:

Job details id	Job name	Company name	Number of jobs	Job type	Skills	Experience	Job last date
1	TCS Java Developer	TCS	50	Full time	java and Java script	Not required	2022-05-10

Below the table, it says 'Showing 1 to 1 of 1 entries' and has navigation buttons for Previous, 1, and Next. At the bottom left, it says 'Developed by -Group Number- 27'.

## College Details:

The screenshot shows a web browser window for 'Campus Tracker' at localhost:8082. The title bar says 'Admin Home Page'. The main content area has a blue header 'Campus Tracker' with a group of people icon. On the left is a sidebar with links: Home, My Profile, Resume, Selected students, Job details, college details, Student Marks, placement officer, Selected in criteria, and Change Password. The main content area is titled 'College Details' with a table showing two records:

College id	College name	Principal	Email id	Contact no	College code
1	ASM IBMR Pune	Pachpande Sir	asmplacement@asmedu.o	9422009207	6973
2	DY Patil	Amit Shrivastav	dypedu@gmail.com	9578461248	4975

Below the table, it says 'Showing 1 to 2 of 2 entries' and has navigation buttons for Previous, 1, and Next. At the bottom left, it says 'Developed by -Group Number- 27'.

# CAMPUS TRACKER

## Student Marks:

Normalization: The Student Log in to the system Can't edit Register no And Name

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/user/students\_marks\_from.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left is a sidebar menu with options like "Home", "My Profile", "Resume", etc. The main content area is titled "Marks Details" and contains a form for "Student Marks" with fields for "Register no.", "Student Name", "SSC Presentage", "HSC Presentage", "Degree", and "Masters". There are "Submit" and "Reset" buttons at the bottom of the form.

## Placement Officer Details:

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/user/placement\_officer\_view.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left is a sidebar menu with options like "Home", "My Profile", "Resume", etc. The main content area is titled "Placement Officer Details" and displays a table with one record. The table columns are "Placement officer id", "Placement officer name", "Qualification", "Email id", and "Contact no". The data shown is: Placement officer id 1, Placement officer name Prof Milind, Qualification PHD, Email id milinddeshkar@asm.edu.org, and Contact no 9503054057. Below the table, it says "Showing 1 to 1 of 1 entries". At the bottom left, it says "Developed by -Group Number- 27".

# CAMPUS TRACKER

## Selected in Criteria: Students

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/user/view\_selected.jsp". The page has a blue header with the "Campus Tracker" logo and a navigation bar for "Student". On the left is a sidebar menu with options like "Home", "My Profile", "Resume", "Selected students", etc. The main content area is titled "view selected" and contains a search form with dropdowns for "Company Name" (set to "SELECT") and "Search" button, and a text input for "Records per page" (set to 10). Below the search form is a table with columns: Student\_no, Register\_no, Student\_name, SSC percentage, HSC percentage, Degree percentage, and Masters percentage. A message "No data available in table" is displayed. At the bottom, it says "Showing 0 to 0 of 0 entries" and has "Previous" and "Next" buttons.

## Selected in Criteria Students

For example: company-TCS

The screenshot shows the same web browser window as the previous one, but with the "Company Name" dropdown set to "TCS". The main content area now displays a table with student data for TCS:

Student_no	Register_no	Student_name	SSC percentage	HSC percentage	Degree percentage	Masters percentage
1	21113	Hrishikesh Mali	66.40	52.30	81.20	70
2	21148	Tejal Patil	83.80	56.63	75.63	75
4	21006	Maheesh Kadapure	65	68	76	65

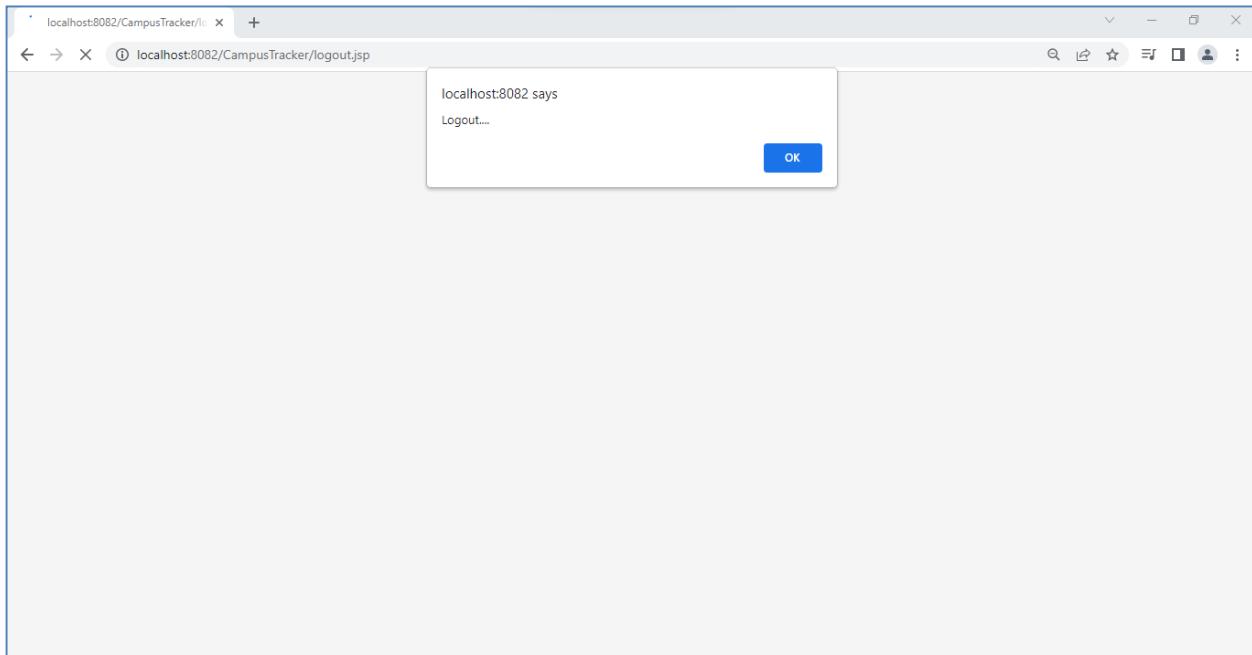
A message "No data available in table" is visible at the bottom of the table area.

# CAMPUS TRACKER

## Change Password :

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/user/change\_password.jsp". The page has a blue header bar with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with options like "Home", "My Profile", "Resume", etc., and a "Change Password" link which is highlighted. The main content area is titled "Change Password" and contains three input fields: "Old password", "New password", and "Confirm password". Below the fields are two buttons: "Submit" (blue) and "Reset" (red). At the bottom left of the content area, it says "Developed by -Group Number- 27".

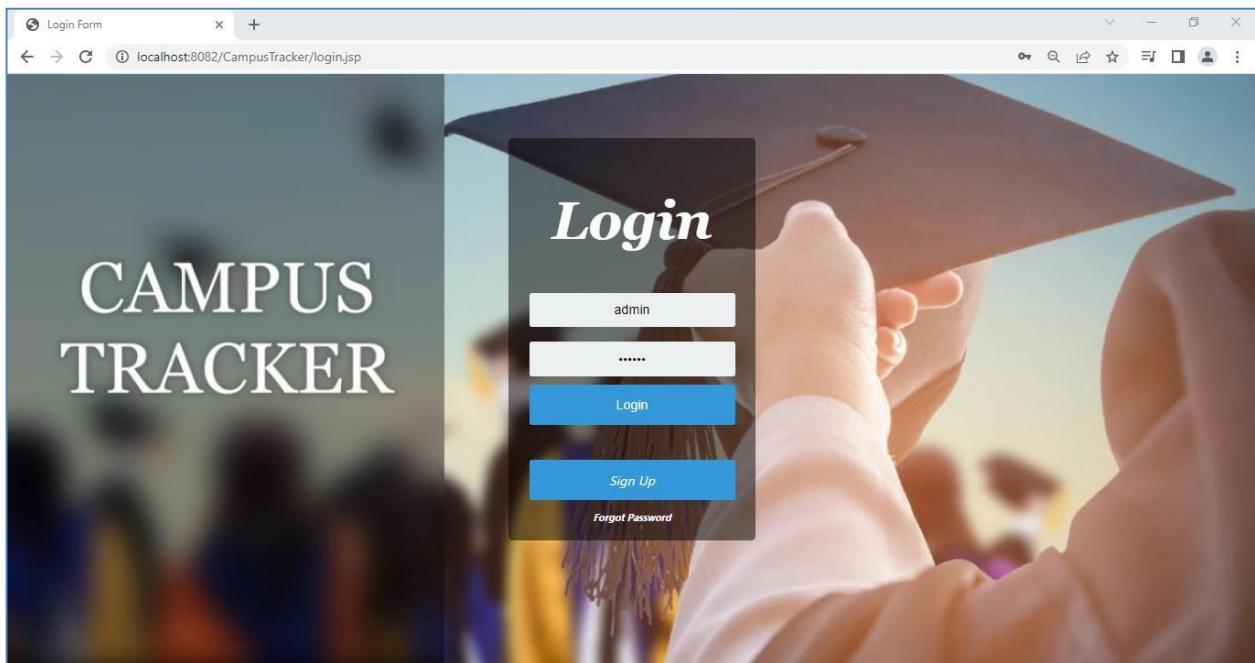
**Log out: Student**



## CAMPUS TRACKER

**Admin Login:**

**For example: Username -admin , Password -100100**



**Admin login**

A screenshot of a web browser showing the 'Admin Home Page' at localhost:8082/CampusTracker/admin/home.jsp. The header features the 'Campus Tracker' logo and a navigation bar with an 'Admin' dropdown. The left sidebar contains a menu with options like 'Home', 'College Details', 'Placement officer', 'Student details', 'Students Resume', 'Company Details', and 'Change Password'. The main content area displays a welcome message 'Wel Come to Admin' and an image of the 'INSTITUTE OF BUSINESS MANAGEMENT &amp; RESEARCH' building, which is affiliated with the University of Pune and approved by AICTE.

# CAMPUS TRACKER

## College Details:

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/admin/college\_details\_view.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with "Home" selected, followed by "College Details", "Placement officer", "Student details", "Students Resume", "Company Details", and "Change Password". The main content area is titled "College Details" and contains a table with two entries. The table columns are: College id, College name, Principal, Email id, Contact no, College code, Delete, and Edit. The entries are:

College id	College name	Principal	Email id	Contact no	College code	Delete	Edit
1	ASM IBMR Pune	Pachpande Sir	asmplacement@asmedu.o	9422009207	6973	<a href="#">Delete</a>	<a href="#">Edit</a>
2	DY Patil	Amit Shrivastav	dypedu@gmail.com	9578461248	4975	<a href="#">Delete</a>	<a href="#">Edit</a>

At the bottom of the content area, it says "Showing 1 to 2 of 2 entries" and has navigation buttons for "Previous", "1", and "Next →". In the footer, it says "Developed by - Group No. 27".

## Function: Add New College

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/admin/college\_details.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with "Home" selected, followed by "College Details", "Placement officer", "Student details", "Students Resume", "Company Details", and "Change Password". The main content area is titled "College Details" and contains a form with fields for "College name", "Principal", "Email id", "Contact no", and "College code". Below the form are "Submit" and "Reset" buttons. In the footer, it says "Developed by - Group No. 27".

# CAMPUS TRACKER

## College Details: Function - Edit , Delete

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/admin/college\_details\_view.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with "Home" selected, followed by "College Details", "Placement officer", "Student details", "Students Resume", "Company Details", and "Change Password". The main content area is titled "College Details" and contains a table with two entries:

College id	College name	Principal	Email id	Contact no	College code	Delete	Edit
1	ASM IBMR Pune	Pachpande Sir	asmplacement@asmedu.o	9422009207	6973	<a href="#">Delete</a>	<a href="#">Edit</a>
2	DY Patil	Amit Shrivastav	dypedu@gmail.com	9578461248	4975	<a href="#">Delete</a>	<a href="#">Edit</a>

Below the table, it says "Showing 1 to 2 of 2 entries" and has navigation buttons for "Previous", "1", and "Next →". At the bottom left, it says "Developed by - Group No. 27".

## Function: Edit

For example: College code - 4975

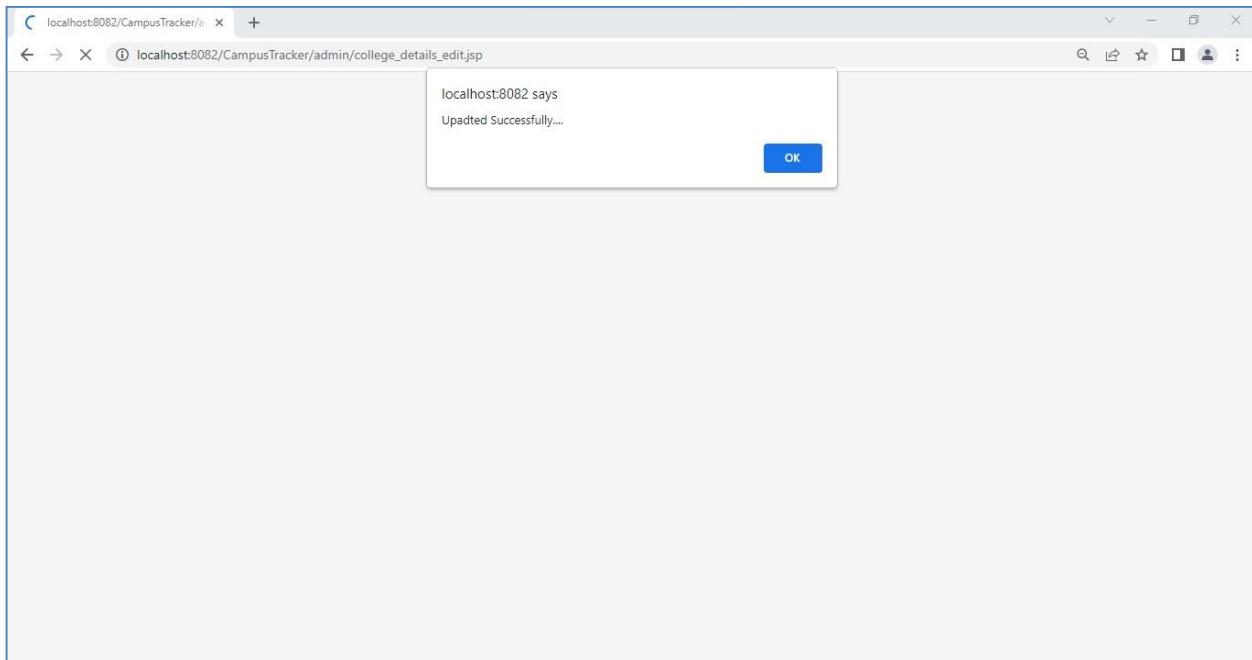
The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/admin/college\_details\_update.jsp?d=2". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with "Home" selected, followed by "College Details", "Placement officer", "Student details", "Students Resume", "Company Details", and "Change Password". The main content area is titled "College Details" and contains a form with the following data:

College name	DY Patil
Principal	Amit
Email id	dypedu@gmail.com
Contact no	9578461249
College code	4975

At the bottom of the form are "Submit" and "Reset" buttons. At the bottom left, it says "Developed by - Group No. 27".

# CAMPUS TRACKER

Updated successfully



Updated College code – 4970

A screenshot of the 'Admin Home Page' from the 'Campus Tracker' application. The page features a blue header with the title 'Campus Tracker' and a navigation menu on the left. The main content area displays a table titled 'College Details' with two rows of data. The table columns are: College id, College name, Principal, Email id, Contact no, College code, Delete, and Edit. The data is as follows:

College id	College name	Principal	Email id	Contact no	College code	Delete	Edit
1	ASM IBMR Pune	Pachpande Sir	asmplacement@asmedu.o	9422009207	6973	<a href="#">Delete</a>	<a href="#">Edit</a>
2	DY Patil	Amit	dypedu@gmail.com	9578461249	4970	<a href="#">Delete</a>	<a href="#">Edit</a>

At the bottom left, it says 'Developed by - Group No. 27'.

# CAMPUS TRACKER

## Function Delete:

For example: delete ASM IBMR , PUNE

The screenshot shows the 'Admin Home Page' at localhost:8082/CampusTracker/admin/college\_details\_view.jsp. The page title is 'Campus Tracker'. On the left, there's a sidebar with a 'Home' button and links for 'College Details', 'Placement officer', 'Student details', 'Students Resume', 'Company Details', and 'Change Password'. The main content area is titled 'College Details' with a 'Records per page:' dropdown set to 10 and a 'Search:' input field. A blue 'Add New College' button is visible. Below is a table with two rows of data:

College id	College name	Principal	Email id	Contact no	College code	Delete	Edit
1	ASM IBMR Pune	Pachpande Sir	asmplacement@asmedu.o	9422009207	6973	<a href="#">Delete</a>	<a href="#">Edit</a>
2	DY Patil	Amit Shrivastav	dypedu@gmail.com	9578461248	4975	<a href="#">Delete</a>	<a href="#">Edit</a>

At the bottom, it says 'Showing 1 to 2 of 2 entries' and has navigation buttons for '← Previous', '1', and 'Next →'.

Developed by - Group No. 27

**College is deleted**

The screenshot shows a confirmation dialog box from localhost:8082/CampusTracker/admin/college\_details\_delete.jsp?id=1. The message inside the box reads: 'localhost:8082 says College is deleted...'. There is an 'OK' button at the bottom right of the dialog.

# CAMPUS TRACKER

After deleted

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/admin/college\_details\_view.jsp". The page features a blue header with the text "Campus Tracker" and a decorative banner of people. On the left, there is a sidebar menu with options like "Home", "College Details", "Placement officer", etc. The main content area is titled "College Details" and contains a table with one row of data. The table columns are "College id", "College name", "Principal", "Email id", "Contact no", "College code", "Delete", and "Edit". The data in the table is:

College id	College name	Principal	Email id	Contact no	College code	Delete	Edit
2	DY Patil	Amit	dypedu@gmail.com	9578461249	4970	<a href="#">Delete</a>	<a href="#">Edit</a>

Below the table, it says "Showing 1 to 1 of 1 entries". At the bottom of the page, it says "Developed by - Group No. 27".

Placement officer details:

Function: Add New Placement Officer

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/admin/placement\_officer\_view.jsp". The page features a blue header with the text "Campus Tracker" and a decorative banner of people. On the left, there is a sidebar menu with options like "Home", "College Details", "Placement officer", etc. The main content area is titled "Placement Officer" and contains a table with one row of data. The table columns are "Placement officer id", "Placement officer name", "Qualification", "Email id", "Contact no", and "Delete". The data in the table is:

Placement officer id	Placement officer name	Qualification	Email id	Contact no	Delete
1	Prof Milind	PHD	milinddeshkar@asm.edu.org	9503054057	<a href="#">Delete</a>

Below the table, it says "Showing 1 to 1 of 1 entries". At the bottom of the page, it says "Developed by - Group No. 27".

# CAMPUS TRACKER

## Function: Add New Placement Officer

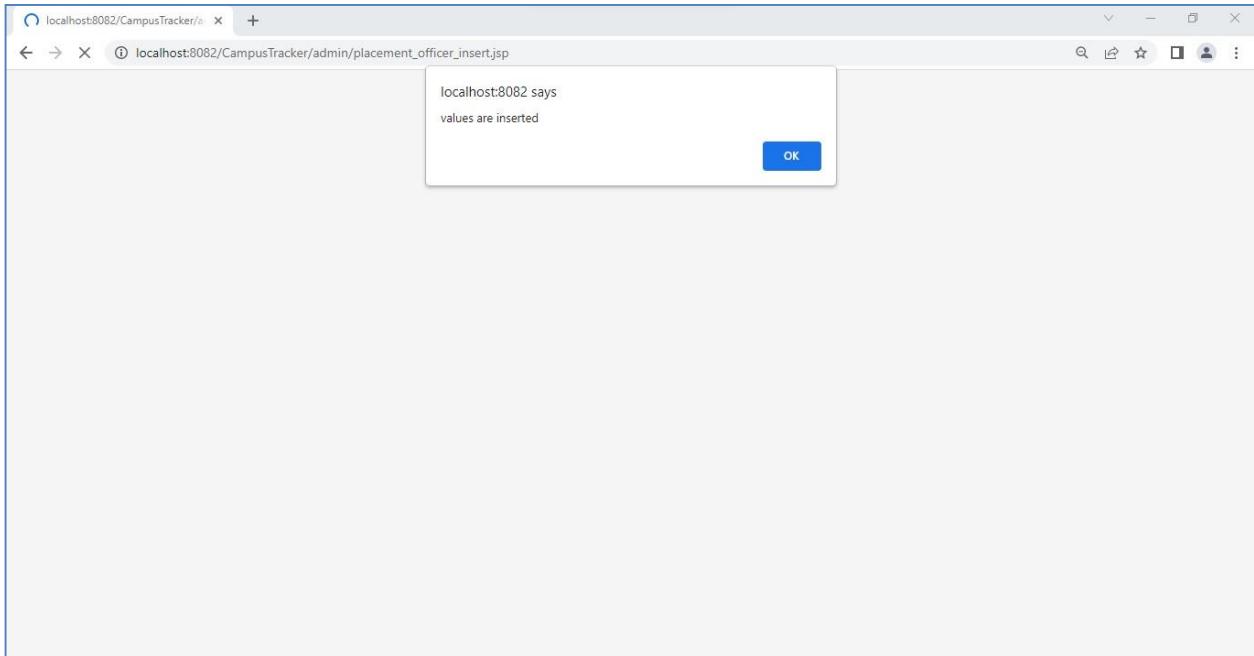
The screenshot shows the 'Admin Home Page' at [localhost:8082/CampusTracker/admin/placement\\_officer.jsp](http://localhost:8082/CampusTracker/admin/placement_officer.jsp). The page title is 'Campus Tracker'. On the left, there's a sidebar menu with 'Home' selected, followed by 'College Details', 'Placement officer', 'Student details', 'Students Resume', 'Company Details', and 'Change Password'. The main content area is titled 'Placement Officer' and contains four input fields: 'Placement officer name' (empty), 'Qualification' (empty), 'Email id' (empty), and 'Contact number' (empty). Below these fields are 'Submit' and 'Reset' buttons. At the bottom left, it says 'Developed by - Group No. 27'.

## Values of new placement officer

The screenshot shows the same 'Admin Home Page' as above, but with different values in the 'Placement Officer' form. The 'Placement officer name' field now contains 'Dr Ashwyn kumar', 'Qualification' is 'PHD', 'Email id' is 'ashwinkumar@gmail.com', and 'Contact number' is '1234356723'. The rest of the page structure remains the same, including the sidebar menu and developer information at the bottom.

# CAMPUS TRACKER

## Values are inserted



## After Values get inserted

A screenshot of the 'Admin Home Page'. The left sidebar has a 'Home' menu with options like 'College Details', 'Placement officer', 'Student details', etc. The main content area is titled 'Placement Officer' and shows a table with two entries. The table columns are 'Placement officer id', 'Placement officer name', 'Qualification', 'Email id', 'Contact no', and 'Delete'. The first row has values: 1, Prof Milind, PHD, milinddeshkar@asm.edu.org, 9503054057, and a 'Delete' link. The second row has values: 2, Dr Ashwyn kumar, PHD, ashwinkumar@gmail.com, 123456723, and a 'Delete' link. There are also buttons for 'Add New Placement Officer', 'Search', and navigation links for 'Previous', '1', and 'Next'.

Developed by - Group No. 27

# CAMPUS TRACKER

## Student Details :

The screenshot shows the 'Admin Home Page' at [localhost:8082/CampusTracker/admin/student\\_details\\_view.jsp](http://localhost:8082/CampusTracker/admin/student_details_view.jsp). The page title is 'Campus Tracker'. On the left, there's a sidebar with a 'Home' button highlighted in blue, and other options like 'College Details', 'Placement officer', 'Student details', 'Students Resume', 'Company Details', and 'Change Password'. The main content area is titled 'Student Details' and contains a table of student records. The table has columns for Student id, Student name, Student full name, Student DOB, Student gender, Student email id, Student contact no, Student reg no, Resume upload, and college\_name. Two entries are listed:

Student id	Student name	Student full name	Student DOB	Student gender	Student email id	Student contact no	Student reg no	Resume upload	college_name
3	Vinayak Lohar	Shivaji	1997-12-04	Male	vinu04@gmail.com	9743949696	21110	No Resume	DY Patil
4	Mahesh Kadapure	Sanjay	1997-08-23	male	mahesh23@gmail.com	9359634785	21006	No Resume	DY Patil

At the bottom, it says 'Showing 1 to 2 of 2 entries' and has navigation buttons for 'Previous', '1', and 'Next -->'. The footer of the page says 'Developed by - Group No. 27'.

## Student details: Function- Add New Student

The screenshot shows the 'Admin Home Page' at [localhost:8082/CampusTracker/admin/student\\_details.jsp](http://localhost:8082/CampusTracker/admin/student_details.jsp). The page title is 'Campus Tracker'. The sidebar is identical to the previous screenshot. The main content area is titled 'Student Details' and contains a form for adding new student information. The fields include: Student name (text input), Student full name (text input), Student DOB (date input), Student gender (radio buttons for male and female), Student email id (text input), Student contact number (text input), Student register number (text input), College id (dropdown menu with 'SELECT' option), and two buttons at the bottom: 'Submit' (blue) and 'Reset' (red).

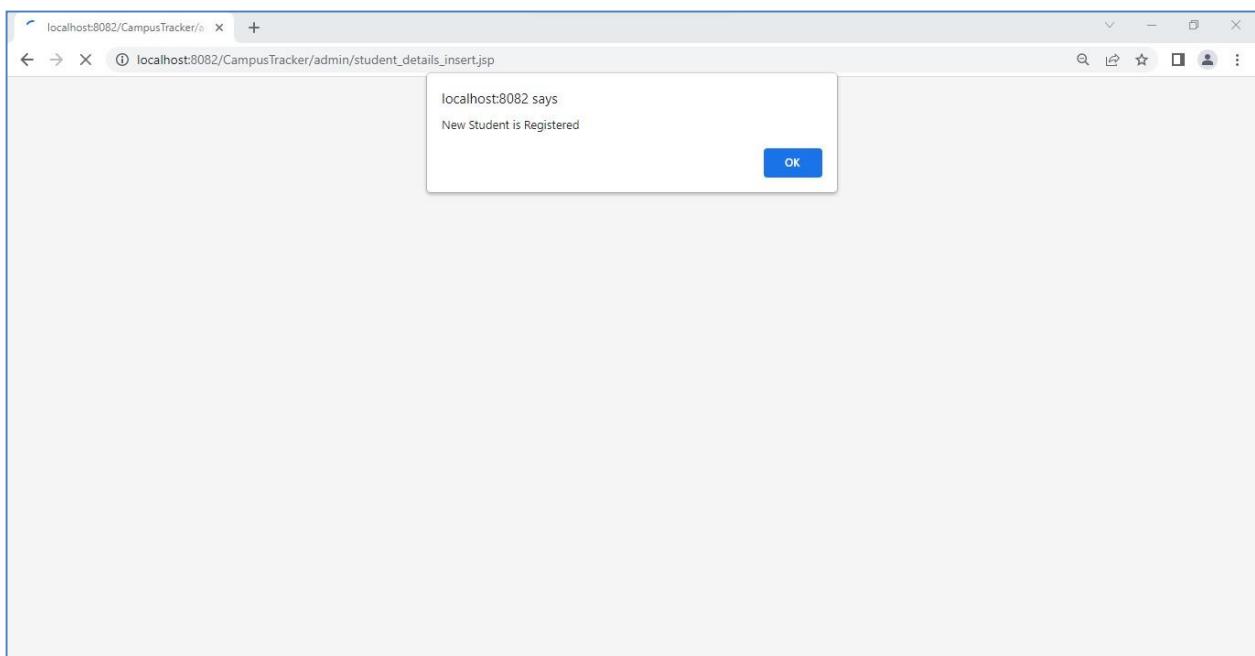
# CAMPUS TRACKER

## Student details: Function- Add New Student

For example, New Student Information

The screenshot shows the 'Admin Home Page' at [localhost:8082/CampusTracker/admin/student\\_details.jsp](http://localhost:8082/CampusTracker/admin/student_details.jsp). The page title is 'Campus Tracker'. On the left, there's a sidebar menu with 'Home' selected, followed by 'College Details', 'Placement officer', 'Student details', 'Students Resume', 'Company Details', and 'Change Password'. The main content area is titled 'Student Details' and contains fields for: Student name (prachi), Student full name (prachi R patil), Student DOB (25-11-2000), Student gender (female), Student email id (prachi29@gmail.com), Student contact number (7564342567), Student register number (21156), and College id (DY Patil). There are 'Submit' and 'Reset' buttons at the bottom.

After submitted , New Student is Registered Successfully



# CAMPUS TRACKER

## Updated New Registered Student:

The screenshot shows the 'Student Details' section of the application. On the left is a navigation sidebar with links for Home, College Details, Placement officer, Student details, Students Resume, Company Details, and Change Password. The main content area has a title 'Student Details' and a button 'Add New Student'. It includes a dropdown for 'Records per page' set to 10, a search bar, and a table with columns: Student id, Student name, Student full name, Student DOB, Student gender, Student email id, Student contact no, Student reg no, Resume upload, and college\_name. The table contains three rows of student data. At the bottom, it says 'Showing 1 to 3 of 3 entries' and has navigation buttons for Previous, Next, and a page number 1.

Student id	Student name	Student full name	Student DOB	Student gender	Student email id	Student contact no	Student reg no	Resume upload	college_name
3	Vinayak Lohar	Shivaji	1997-12-04	Male	vinu04@gmail.com	9743949696	21110	No Resume	DY Patil
4	Mahesh Kadapure	Sanjay	1997-08-23	male	mahesh23@gmail.com	9359634785	21006	No Resume	DY Patil
5	prachi	prachi R patil	2000-11-25	female	prachi29@gmail.com	7564342567	21156	No Resume	DY Patil

## Students Resume:

The screenshot shows the 'Resume' section of the application. The left sidebar is identical to the previous screenshot. The main content area has a title 'Resume' and a table with columns: Student id, Student Name, Reg No, and Resume. The table contains five rows of student data, each with a 'Download' button. At the bottom, it says 'Showing 1 to 5 of 5 entries' and has navigation buttons for Previous, Next, and a page number 1.

Student id	Student Name	Reg No	Resume
1	Hrishikesh Mali	21113	<button>Download</button>
2	Tejal Patil	21148	<button>Download</button>
3	Vinayak Lohar	21110	<button>Download</button>
4	Mahesh Kadapure	21006	<button>Download</button>
5	prachi	21156	<button>Download</button>

## CAMPUS TRACKER

If Resume is Not Uploaded: Pop Up Notification- “Please Upload Resume”

The screenshot shows the 'Admin Home Page' at [localhost:8082/CampusTracker/admin/resume\\_view.jsp](http://localhost:8082/CampusTracker/admin/resume_view.jsp). The page title is 'Campus Tracker'. On the left, there is a sidebar menu with options: Home, College Details, Placement officer, Student details, Students Resume, Company Details, and Change Password. The main content area is titled 'Resume' and displays a table of student records. The table has columns for Student id, Student Name, Reg No, and Resume. The first row shows Hirshikesh Mali with Reg No 21113 and a 'Please Upload Resume' button. The other four students have 'Download' buttons. At the bottom, it says 'Showing 1 to 5 of 5 entries' and has navigation buttons for Previous, 1, and Next.

Student id	Student Name	Reg No	Resume
1	Hirshikesh Mali	21113	Please Upload Resume
2	Tejal Patil	21148	Download
3	Vinayak Lohar	21110	Download
4	Mahesh Kadapure	21006	Download
5	prachi	21156	Download

For Admin : If Resume is Not Updated : Pop Up Notification- “Student not updated resume”

The screenshot shows the same 'Admin Home Page' as the previous one, but with a modal dialog box in the center. The dialog box has a blue header bar with the text 'localhost:8082 says' and a message below it: 'Student not updated resume'. It has an 'OK' button at the bottom right. The rest of the page is identical to the first screenshot, showing the 'Resume' table with student records and their download/upload status.

Student id	Student Name	Reg No	Resume
1	Hirshikesh Mali	21113	Please Upload Resume
2	Tejal Patil	21148	Download
3	Vinayak Lohar	21110	Download
4	Mahesh Kadapure	21006	Download
5	prachi	21156	Download

# CAMPUS TRACKER

## Downloaded Resume :

The screenshot shows the 'Admin Home Page' at [localhost:8082/CampusTracker/admin/resume\\_view.jsp](http://localhost:8082/CampusTracker/admin/resume_view.jsp). The page title is 'Campus Tracker'. On the left, there's a navigation menu with 'Home' selected, followed by 'College Details', 'Placement officer', 'Student details', 'Students Resume', 'Company Details', and 'Change Password'. The main content area is titled 'Resume' with a search bar and a dropdown for 'Records per page: 10'. It lists five student entries:

Student Id	Student Name	Reg No	Resume
1	Hrishikesh Mali	21113	<a href="#">Please Upload Resume</a>
2	Tejal Patil	21148	<a href="#">Download</a>
3	Vinayak Lohar	21110	<a href="#">Download</a>
4	Mahesh Kadapure	21006	<a href="#">Download</a>
5	prachi	21156	<a href="#">Download</a>

Below the table, it says 'Showing 1 to 5 of 5 entries'. At the bottom, there are links for 'Previous', '1', 'Next →', 'Show all', and a close button 'X'. A file manager window is open at the bottom, showing two files: 'Hrishikesh Mali (1).docx' and 'Hrishikesh Mali (....docx)', with the second one being removed.

## Company Details:

The screenshot shows the 'Admin Home Page' at [localhost:8082/CampusTracker/admin/view\\_company\\_details.jsp](http://localhost:8082/CampusTracker/admin/view_company_details.jsp). The page title is 'Campus Tracker'. The left navigation menu is identical to the previous screenshot. The main content area is titled 'Company Details' with a search bar and a dropdown for 'Records per page: 10'. It lists one company entry:

C-ID	Company Name	Company Address	Domain Name	Work Field	E-Mail	Contact No	C Code	Delete	Update
1	TCS	Pune- Hinjewadi Phase 3	Banking	java	tcsfreshers@tcs.com	874579426	45876	<a href="#">Delete</a>	<a href="#">Update</a>

Below the table, it says 'Showing 1 to 1 of 1 entries'. At the bottom, there are links for 'Previous', '1', 'Next →', and a close button 'X'. The footer of the page says 'Developed by - Group No. 27'.

# CAMPUS TRACKER

## Company Details: Function- Add New Company Details

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/admin/company\_details\_form.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with "Home" selected, followed by "College Details", "Placement officer", "Student details", "Students Resume", "Company Details", and "Change Password". The main content area is titled "Company Details" and contains a form with fields for Company Name, Company Address, Domain name, Work field, Email Id, Contact No, and Company code. There are "Submit" and "Reset" buttons at the bottom.

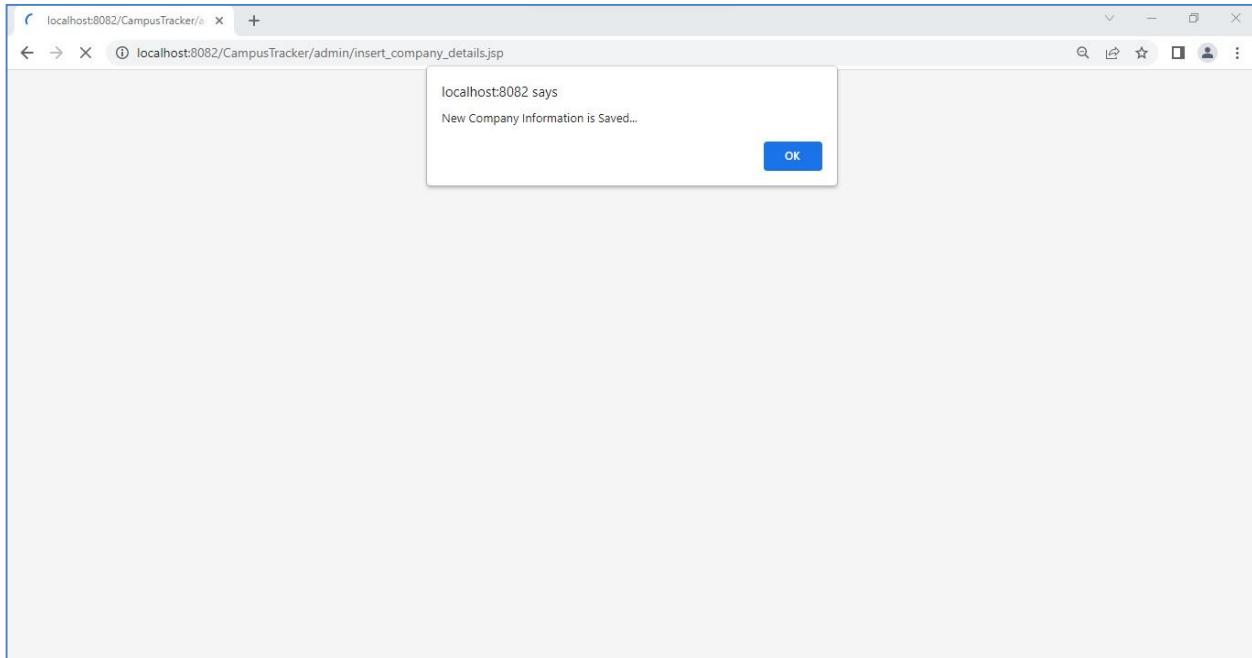
## Function- Add New Company Details

For example

The screenshot shows the same web browser window as the previous one, but with the "Company Details" form populated with data. The fields are filled as follows: Company Name (Infosys), Company Address (Magarpatta), Domain name (support), Work field (business analyst), Email Id (infosys@gmail.com), Contact No (2341567832), and Company code (123456). The "Submit" and "Reset" buttons are visible at the bottom of the form.

# CAMPUS TRACKER

## New Company Information Is Saved Sucessfully



## Updated Company Information

A screenshot of the 'Admin Home Page' titled 'Campus Tracker'. The left sidebar has a 'Home' tab selected, along with other options like 'College Details', 'Placement officer', 'Student details', 'Students Resume', 'Company Details', and 'Change Password'. The main content area is titled 'Company Details' and contains a table with two entries. The table columns are C-ID, Company Name, Company Address, Domain Name, Work Field, E-Mail, Contact No, C Code, Delete, and Update. The first entry (C-ID 1) is for TCS with details: Company Name - TCS, Company Address - Pune- Hinjewadi Phase 3, Domain Name - Banking, Work Field - java, E-Mail - tcsfreshers@tcs.com, Contact No - 874579426, C Code - 45876, and buttons for Delete and Update. The second entry (C-ID 2) is for Infosys with details: Company Name - Infosys, Company Address - Magarpatta, Domain Name - support, Work Field - business analyst, E-Mail - infosys@gmail.com, Contact No - 2341567832, C Code - 123456, and buttons for Delete and Update. At the bottom of the table, it says 'Showing 1 to 2 of 2 entries' and has navigation buttons for Previous, 1, and Next.

# CAMPUS TRACKER

## Company details: Function- Update

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/admin/view\_company\_details.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with options like "Home", "College Details", "Placement officer", "Student details", "Students Resume", "Company Details", and "Change Password". The main content area is titled "Company Details" and contains a table with two entries. The table columns are C-ID, Company Name, Company Address, Domain Name, Work Field, E-Mail, Contact No, and C Code. The first entry is for TCS with C-ID 1, and the second is for Infosys with C-ID 3. There are "Delete" and "Update" buttons for each row. At the bottom, it says "Showing 1 to 2 of 2 entries".

C-ID	Company Name	Company Address	Domain Name	Work Field	E-Mail	Contact No	C Code	Delete	Update
1	TCS	Pune- Hinjewadi Phase 3	Banking	java	tcsfreshers@tcs.com	874579426	45876	<a href="#">Delete</a>	<a href="#">Update</a>
3	Infosys	magarpatta	support	business analyst	infosys@gmail.com	2341567832	123456	<a href="#">Delete</a>	<a href="#">Update</a>

## Company details: Function- Update

for ex. Work Field- business analyst

This screenshot is identical to the one above, showing the "Company Details" page with the same data and layout. It includes the sidebar menu, the table with two entries, and the pagination information at the bottom.

C-ID	Company Name	Company Address	Domain Name	Work Field	E-Mail	Contact No	C Code	Delete	Update
1	TCS	Pune- Hinjewadi Phase 3	Banking	java	tcsfreshers@tcs.com	874579426	45876	<a href="#">Delete</a>	<a href="#">Update</a>
2	Infosys	Magarpatta	support	business analyst	infosys@gmail.com	2341567832	123456	<a href="#">Delete</a>	<a href="#">Update</a>

# CAMPUS TRACKER

## Company details : Function- Updated Field

for ex. Work Field- Cloud business analyst

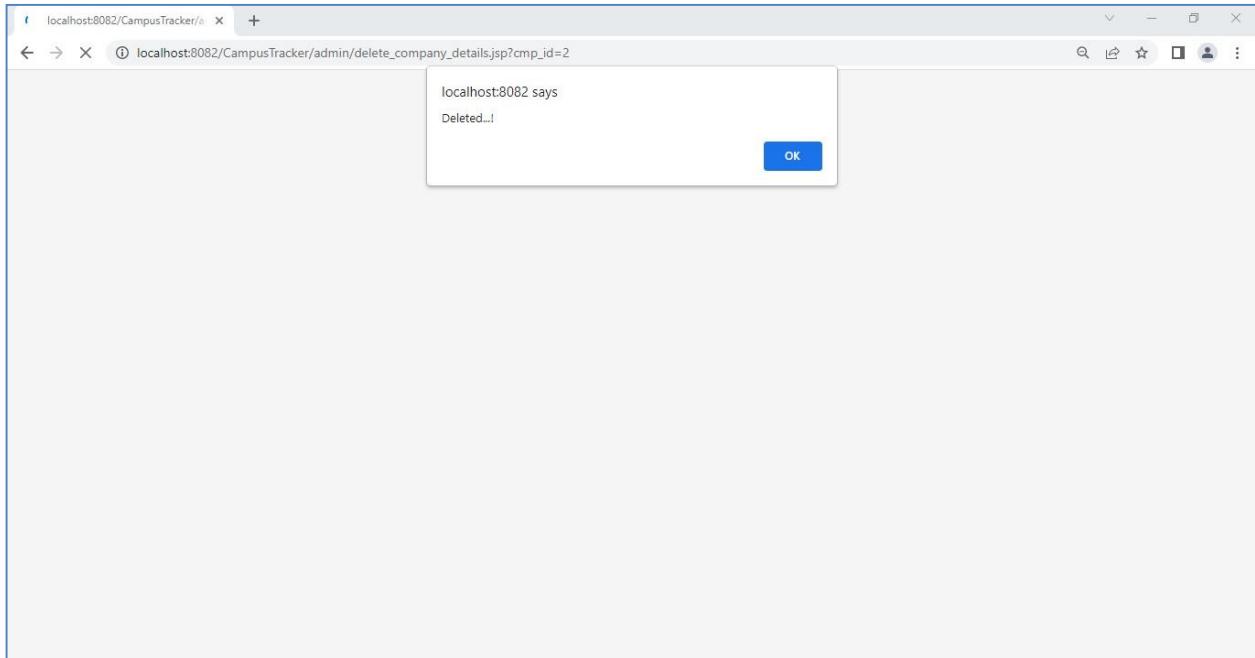
The screenshot shows a web browser window for 'Admin Home Page' at 'localhost:8082/CampusTracker/admin/view\_company\_details.jsp'. The title bar says 'Campus Tracker' with a group of people icon. The left sidebar has a 'Home' button and links for 'College Details', 'Placement officer', 'Student details', 'Students Resume', 'Company Details', and 'Change Password'. The main content area is titled 'Company Details' with a table. The table has columns: C-ID, Company Name, Company Address, Domain Name, Work Field, E-Mail, Contact No, C Code, Delete, and Update. It shows two entries: TCS (C-ID 1) with address 'Pune- Hinjewadi Phase 3', domain 'tcsfreshers@tcs.com', work field 'java', and contact '874579426'; and Infosys (C-ID 2) with address 'Magarpatta', domain 'infosys@gmail.com', work field 'cloud business analyst', and contact '2341567832'. A message at the bottom says 'Showing 1 to 2 of 2 entries'. A footer note says 'Developed by - Group No. 27'.

## Company details : Function- Delete

The screenshot shows the same web browser window as the previous one, but the 'Work Field' column for both entries is now empty. The table structure remains the same, but the data for 'Work Field' is missing. The rest of the interface, including the sidebar and footer, is identical to the previous screenshot.

# CAMPUS TRACKER

## Company Information Deleted Successfully



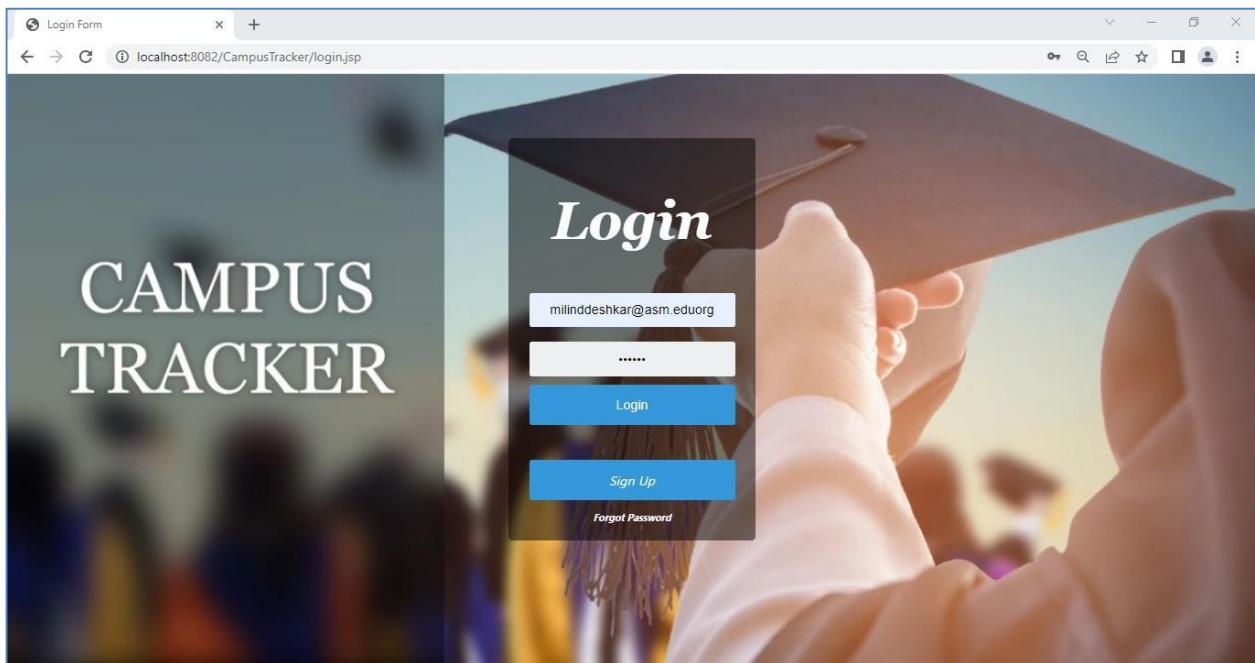
## After Deleted for Remaining Values

A screenshot of the 'Admin Home Page' of the Campus Tracker application. The title bar says 'Campus Tracker'. On the left, there is a sidebar menu with options: Home (selected), College Details, Placement officer, Student details, Students Resume, Company Details, and Change Password. The main content area is titled 'Company Details' and contains a table with one row of data. The table columns are: C-ID, Company Name, Company Address, Domain Name, Work Field, E-Mail, Contact No, C Code, Delete, and Update. The data in the table is: 1, TCS, Pune- Hinjewadi Phase 3, Banking, java, tcsfreshers@tcs.com, 874579426, 45876, a 'Delete' link, and an 'Update' link. At the bottom of the page, it says 'Showing 1 to 1 of 1 entries' and has navigation links for '← Previous', '1', and 'Next →'. The footer of the page says 'Developed by - Group No. 27'.

## CAMPUS TRACKER

### Placement Officer Login :

For example: Username - milinddeshkar@asm.eduorg , Password -100100



### Placement Officer Dashboard/Report

Normalization: The Placement Officer Which Log into the System that we can see in Placement Officer Profile

A screenshot of a web browser window titled "Admin Home Page". The URL is "localhost:8082/CampusTracker/pl\_officer/placement\_officer\_view.jsp". The page has a blue header with the "Campus Tracker" logo and a navigation bar on the left with options like "Home", "College details", "Campus Details", etc. The main content area shows a table titled "Placement Officer" with one record. The table columns are "Placement officer id", "Placement officer name", "Qualification", "Email id", and "Contact no". The data is: Placement officer id 1, Placement officer name Prof Milind, Qualification PHD, Email id milinddeshkar@asm.edu.org, Contact no 9503054057. At the bottom, it says "Showing 1 to 1 of 1 entries". On the right, there's a user menu with "Profile" and "Logout". A small note at the bottom left says "Developed by - Group No. 27".

# CAMPUS TRACKER

## Placement Officer Login



## College Details:

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/pl\_officer/college\_details\_view.jsp". The page has a blue header bar with the title "Campus Tracker" and a decorative graphic of people walking. A sidebar on the left contains a navigation menu with options like "Home", "College details", "Campus Details", "Student details", "Resume", "Shortlisted students", "Selected students", and "Change Password". The main content area is titled "College Details" and displays a table with college information. The table has columns for "College id", "College name", "Principal", "Email id", "Contact no", and "College code". There is also a "Delete" button for each row. The table shows one entry: College id 2, College name DY Patil, Principal Amit, Email id dypedu@gmail.com, Contact no 9578461249, and College code 4970. At the bottom, there is a message "Showing 1 to 1 of 1 entries" and navigation links for "Previous", "1", and "Next".

# CAMPUS TRACKER

## Campus Details:

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/pl\_officer/view\_company\_details.jsp". The page has a blue header with the title "Campus Tracker" and a group of people icon. On the left, there's a sidebar menu with "Home" selected, followed by "College details", "Campus Details", "Student details", "Resume", "Shortlisted students", "Selected students", and "Change Password". The main content area is titled "Company Details" and contains a table with two entries. The table columns are C-ID, Company Name, Company Address, Domain Name, Work Field, E-Mail, Contact No, and C Code. The entries are:

C-ID	Company Name	Company Address	Domain Name	Work Field	E-Mail	Contact No	C Code
1	TCS	Pune- Hinjewadi Phase 3	Banking	java	tcsfreshers@tcs.com	874579426	45876
3	Infosys	magarpatta	support	business analyst	infosys@gmail.com	2341567832	123456

At the bottom, it says "Showing 1 to 2 of 2 entries" and has navigation buttons for "Previous", "1", and "Next →".

## Student Details:

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/pl\_officer/student\_details\_view.jsp". The page has a blue header with the title "Campus Tracker" and a group of people icon. On the left, there's a sidebar menu with "Home" selected, followed by "College details", "Campus Details", "Student details", "Resume", "Shortlisted students", "Selected students", and "Change Password". The main content area is titled "Student Details" and contains a table with three entries. The table columns are Student id, Student name, Student full name, Student DOB, Student gender, Student email id, Student contact no, Student reg no, Resume upload, and college\_name. The entries are:

Student id	Student name	Student full name	Student DOB	Student gender	Student email id	Student contact no	Student reg no	Resume upload	college_name
3	Vinayak Lohar	Shivaji	1997-12-04	Male	vinu04@gmail.com	9743949696	21110	No Resume	DY Patil
4	Mahesh Kadapure	Sanjay	1997-08-23	male	mahesh23@gmail.com	9359634785	21006	No Resume	DY Patil
5	prachi	prachi R patil	2000-11-25	female	prachi29@gmail.com	7564342567	21156	No Resume	DY Patil

At the bottom, it says "Showing 1 to 3 of 3 entries" and has navigation buttons for "← Previous", "1", and "Next →".

# CAMPUS TRACKER

## Resume Information: For Placement officer

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/pl\_officer/resume\_view.jsp". The page features a blue header with the "Campus Tracker" logo and a group of people icon. A sidebar on the left contains links: Home, College details, Campus Details, Student details, Resume, Shortlisted students, Selected students, and Change Password. The main content area is titled "resume" and includes a search bar and a dropdown for "Records per page" set to 10. A table lists student information with columns: Student id, Student Name, Reg No, and Resume (with a "Download" button). The table shows 5 entries. At the bottom, there is a message "Showing 1 to 5 of 5 entries" and navigation buttons for Previous, Next, and a page number 1.

Student id	Student Name	Reg No	Resume
1	Hrishikesh Mali	21113	<button>Download</button>
2	Tejal Patil	21148	<button>Download</button>
3	Vinayak Lohar	21110	<button>Download</button>
4	Mahesh Kadapure	21006	<button>Download</button>
5	prachi	21156	<button>Download</button>

## Shortlisted Students: Under Company –TCS

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/pl\_officer/view\_selected.jsp". The page features a blue header with the "Campus Tracker" logo and a group of people icon. A sidebar on the left contains links: Home, College details, Campus Details, Student details, Resume, Shortlisted students, Selected students, and Change Password. The main content area is titled "view selected" and includes a dropdown for "Company Name" with options "SELECT" and "Ok", and a search bar. A dropdown for "Records per page" is set to 10. A table lists student information with columns: Student\_no, Register\_no, Student\_name, SSLC percentage, PUC percentage, Degree percentage, and Masters percentage. The table shows 3 entries. At the bottom, there is a message "Showing 0 to 0 of 0 entries" and navigation buttons for Previous, Next, and a page number 1.

Student_no	Register_no	Student_name	SSLC percentage	PUC percentage	Degree percentage	Masters percentage
1	21113	Hrishikesh Mali	66.40	52.30	81.20	70
2	21148	Tejal Patil	83.80	56.63	75.63	75
4	21006	Mahesh Kadapure	65	68	76	65

# CAMPUS TRACKER

## Selected Students - Under Placement Officer

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/pl\_officer/selected\_student\_view.jsp". The page has a blue header with the title "Campus Tracker" and a group of people icon. On the left is a sidebar with links: Home, College details, Campus Details, Student details, Resume, Shortlisted students, Selected students, and Change Password. The main content area is titled "selected students" and shows a table with one entry:

Selected student id	Student Reg no.	Placement officer id	Job details id	Selected date	placement_officer_name
1	21148	1	1	2022-02-15	Prof Milind

Below the table, it says "Showing 1 to 1 of 1 entries" and has navigation buttons for Previous, Next, and a page number 1. At the bottom left, it says "Developed by - Group No. 27".

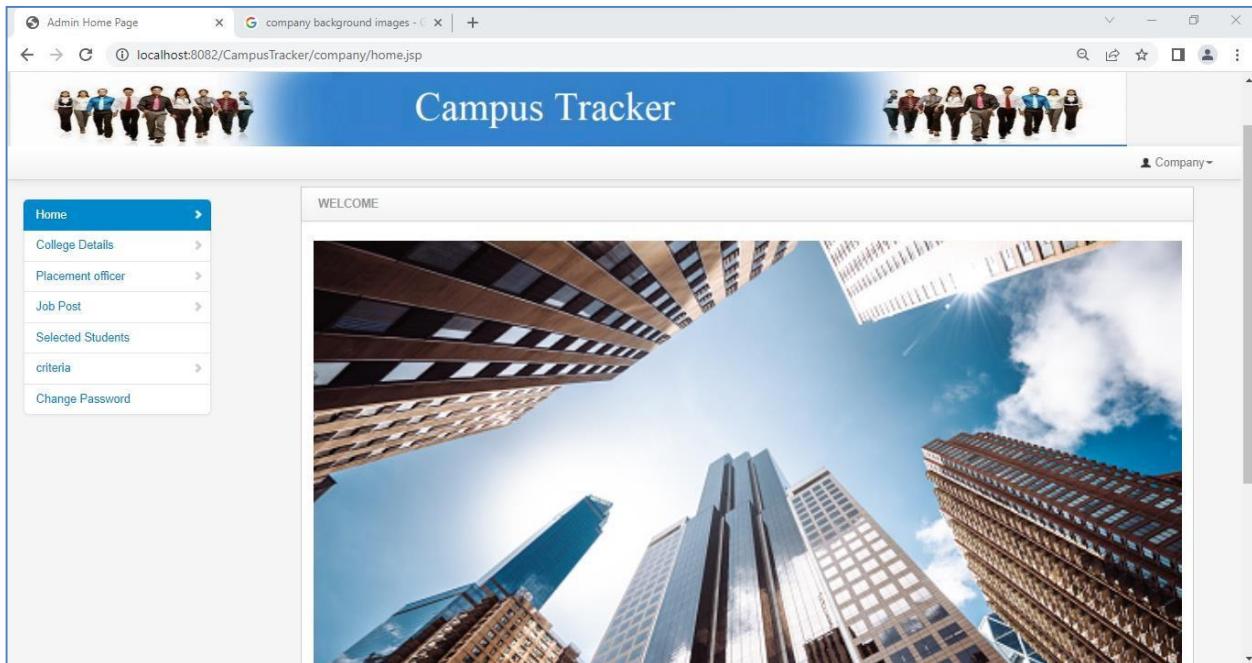
## Company Login:

For example: Username - TCS , Password -100100

The screenshot shows a web browser window titled "Login Form" with the URL "localhost:8082/CampusTracker/login.jsp". The page features a background image of a person in a graduation cap. In the center is a login dialog box with the title "Login". It contains three input fields: the first is pre-filled with "TCS", the second is a password field with dots, and the third is a blue "Login" button. Below the login button are two more buttons: "Sign Up" and "Forgot Password".

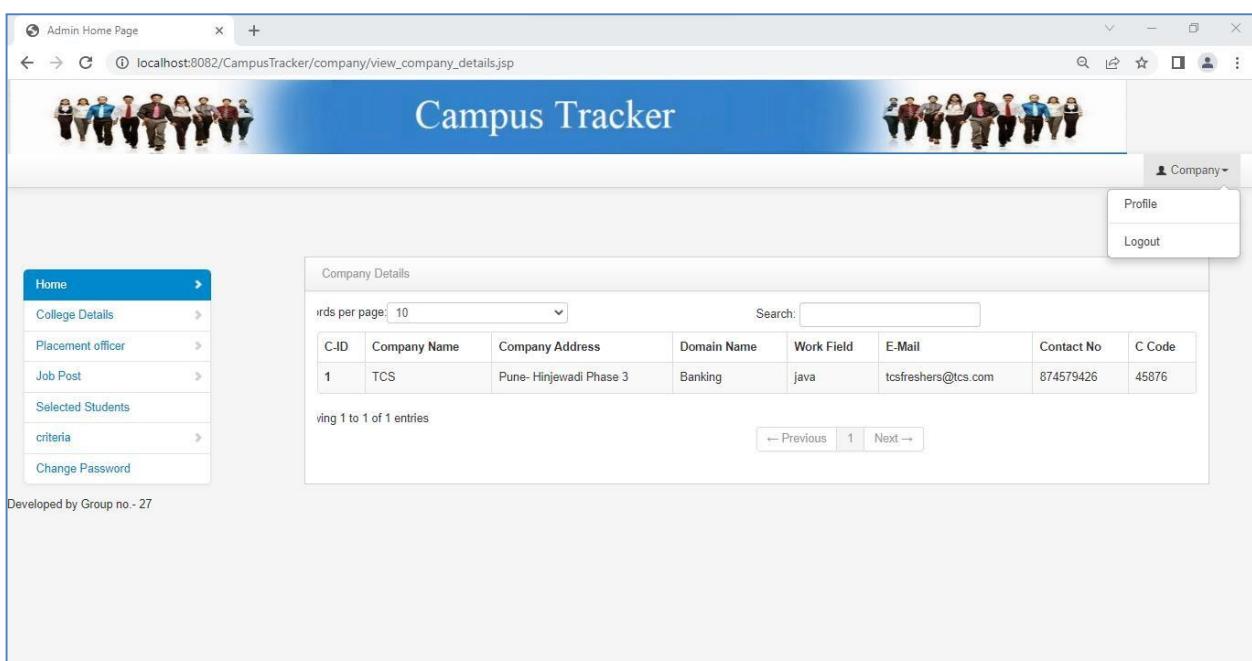
# CAMPUS TRACKER

## Company Login :



## Company Dashboard/Report

**Normalization: The Company Which Log into the System that we can see in Company Profile**



C-ID	Company Name	Company Address	Domain Name	Work Field	E-Mail	Contact No	C Code
1	TCS	Pune- Hinjewadi Phase 3	Banking	java	tcsfreshers@tcs.com	874579426	45876

Showing 1 to 1 of 1 entries

[← Previous](#) [1](#) [Next →](#)

Developed by Group no.- 27

## CAMPUS TRACKER

### College Details:

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/company/college\_details\_view.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with "Home" selected, followed by "College Details", "Placement officer", "Job Post", "Selected Students", "criteria", and "Change Password". The main content area is titled "College Details" and contains a table with one row of data. The table columns are "College id", "College name", "Principal", "Email id", "Contact no", "College code", and "Delete". The data in the table is: College id 2, College name DY Patil, Principal Amit, Email id dypedu@gmail.com, Contact no 9578461249, College code 4970, and Delete button. Below the table, it says "Showing 1 to 1 of 1 entries" and has "Previous" and "Next" buttons. At the bottom left, it says "Developed by Group no.- 27".

### Company has Access to Add New College

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/company/college\_details.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with "Home" selected, followed by "College Details", "Placement officer", "Job Post", "Selected Students", "criteria", and "Change Password". The main content area is titled "College Details" and contains a form with fields for "College name", "Principal", "Email id", "Contact no", and "College code". Below the form are "Submit" and "Reset" buttons. At the bottom left, it says "Developed by Group no.- 27".

# CAMPUS TRACKER

## Placement Officer Details :

The screenshot shows a web application window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/company/placement\_officer\_view.jsp". The main content area is titled "Placement Officer" and displays a table with two rows of data. The columns are "Placement officer id", "Placement officer name", "Qualification", "Email id", "Contact no", and "Delete". The data is as follows:

Placement officer id	Placement officer name	Qualification	Email id	Contact no	Delete
1	Prof Milind	PHD	milinddeshkar@asm.eduorg	9503054057	<a href="#">Delete</a>
2	Dr Ashwini Kumar	PHD	ashwinkumar@gmail.com	123456723	<a href="#">Delete</a>

Below the table, there is a message "Showing 1 to 2 of 2 entries" and a navigation bar with links for "← Previous", "1", and "Next →". On the left sidebar, under the "Home" tab, the "Placement officer" option is selected. At the bottom left, it says "Developed by Group no.- 27".

## Job Post Function- Add New Job Post

The screenshot shows a web application window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/company/job\_detail\_view.jsp". The main content area is titled "Job Details" and displays a table with one row of data. The columns are "Job id", "Technical / Non-technical", "Company name", "Number of openings", "Job type", "Skills", "Experience", "Last Apply date", and "Delete". The data is as follows:

Job id	Technical / Non-technical	Company name	Number of openings	Job type	Skills	Experience	Last Apply date	Delete
1	TCS Java Developer	TCS	50	Full time	java and Java script	Not required	2022-05-10	<a href="#">Delete</a>

Below the table, there is a message "Showing 1 to 1 of 1 entries" and a navigation bar with links for "← Previous", "1", and "Next →". On the left sidebar, under the "Home" tab, the "Job Post" option is selected. At the bottom left, it says "Developed by Group no.- 27".

# CAMPUS TRACKER

## Company Details: Function- Add New Job Post

Normalization: The Company Log into the system Can't edit Company Name

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/company/job\_details.jsp". The page has a blue header bar with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with the following items: Home (selected), College Details, Placement officer, Job Post, Selected Students, criteria, and Change Password. The main content area is titled "Job Details" and contains the following fields:

Job Name	<input type="text"/>
Company Name	<input type="text" value="TCS"/>
Number of jobs	<input type="text"/>
Job Type	<input type="text"/>
Skills	<input type="text"/>
Experience	<input type="text"/>
Job last date	<input type="text"/>

At the bottom right of the form are two buttons: "Submit" and "Reset".

## Values:

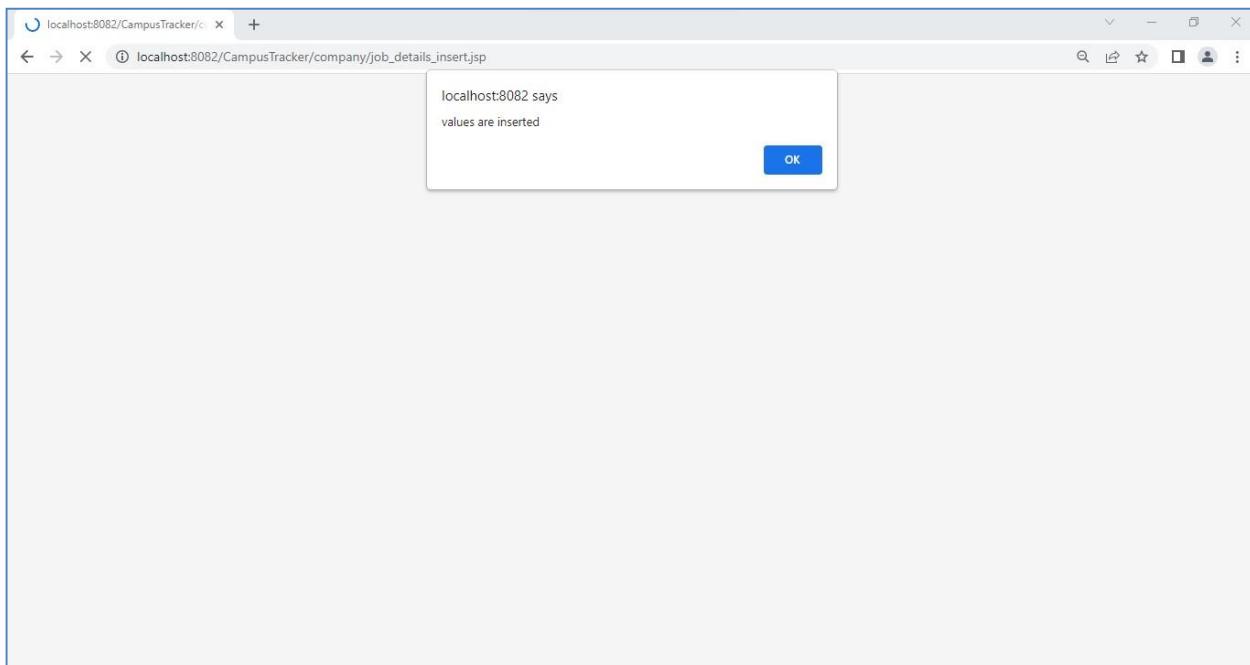
The screenshot shows the same web browser window and application layout as the previous screenshot. The sidebar menu is identical. The "Job Details" form now contains the following values:

Job Name	<input type="text" value="Non Technical"/>
Company Name	<input type="text" value="TCS"/>
Number of jobs	<input type="text" value="35"/>
Job Type	<input type="text" value="support"/>
Skills	<input type="text" value="c cpp"/>
Experience	<input type="text" value="Fresher"/>
Job last date	<input type="text" value="2022-02-02"/>

Again, the "Submit" and "Reset" buttons are at the bottom right.

# CAMPUS TRACKER

Values are inserted Successfully



Updated Values:

A screenshot of the 'Campus Tracker' application. The top navigation bar includes 'Admin Home Page', 'Company', and a user icon. The main content area has a header 'Job Details' with a 'Add New' button. Below it is a table showing job details. The table has columns: Job id, Technical / Non-technical, Company name, Number of openings, Job type, Skills, Experience, Last Apply date, and Delete. Two rows of data are visible:

Job id	Technical / Non-technical	Company name	Number of openings	Job type	Skills	Experience	Last Apply date	Delete
1	TCS Java Developer	TCS	50	Full time	java and Java script	Not required	2022-05-10	<a href="#">Delete</a>
2	Non Technical	TCS	35	support	c cpp	Fresher	2022-02-02	<a href="#">Delete</a>

At the bottom left, a sidebar menu includes 'Home', 'College Details', 'Placement officer', 'Job Post', 'Selected Students', 'criteria', and 'Change Password'. At the bottom center, it says 'Showing 1 to 2 of 2 entries' and has navigation buttons for 'Previous', '1', and 'Next →'. At the very bottom, it says 'Developed by Group no.- 27'.

# CAMPUS TRACKER

## Company Details: Function – Delete

The screenshot shows a web browser window for 'Campus Tracker' at localhost:8082/CampusTracker/company/job\_detail\_view.jsp. The page has a blue header with the title 'Campus Tracker' and a group of people icon. A sidebar on the left contains links: Home, College Details, Placement officer, Job Post, Selected Students, criteria, and Change Password. The main content area is titled 'Job Details' and displays a table of job records. The table has columns: Job id, Technical / Non-technical, Company name, Number of openings, Job type, Skills, Experience, Last Apply date, and Delete. Two entries are listed:

Job id	Technical / Non-technical	Company name	Number of openings	Job type	Skills	Experience	Last Apply date	Delete
1	TCS Java Developer	TCS	50	Full time	java and Java script	Not required	2022-05-10	<a href="#">Delete</a>
2	Non Technical	TCS	35	support	c cpp	Fresher	2022-02-02	<a href="#">Delete</a>

At the bottom, it says 'Showing 1 to 2 of 2 entries' and has navigation buttons for Previous, 1, and Next. A footer note 'Developed by Group no.- 27' is visible.

**Values Deleted Successfully**

A confirmation dialog box is displayed in the center of the screen, showing the message 'localhost:8082 says Value is deleted...' with an 'OK' button.

# CAMPUS TRACKER

## After Deleted Remaining Values

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/company/view\_criteria.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with options: Home, College Details, Placement officer, Job Post, Selected Students, criteria, and Change Password. The main content area is titled "View Criteria" and contains a table with one row of data:

Id	Company Name	SSC	HSC	DEGREE	Masters	Delete
1	TCS	60	50	75	70	<a href="#">Delete</a>

Below the table, a message says "No data available in table". At the bottom, there are links for "Showing 0 to 0 of 0 entries", "← Previous", and "Next →".

At the very bottom of the page, it says "Developed by Group no - 27".

## Criteria: Company criteria

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/company/selection\_criteria\_form.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with options: Home, College Details, Placement officer, Job Post, Selected Students, criteria, and Change Password. The main content area is titled "Campus Details" and contains a form titled "Enter selection Criteria" with fields for Company, SSC, HSC, Degree, and Master. The "Company" field is populated with "TCS". Below the form are two buttons: "Submit" (green) and "Reset" (red).

At the bottom of the page, it says "Developed by Group no - 27".

# CAMPUS TRACKER

## Criteria Values

Admin Home Page

localhost:8082/CampusTracker/company/selection\_criteria\_form.jsp

Campus Tracker

Company

Home

College Details

Placement officer

Job Post

Selected Students

criteria

Change Password

Campus Details

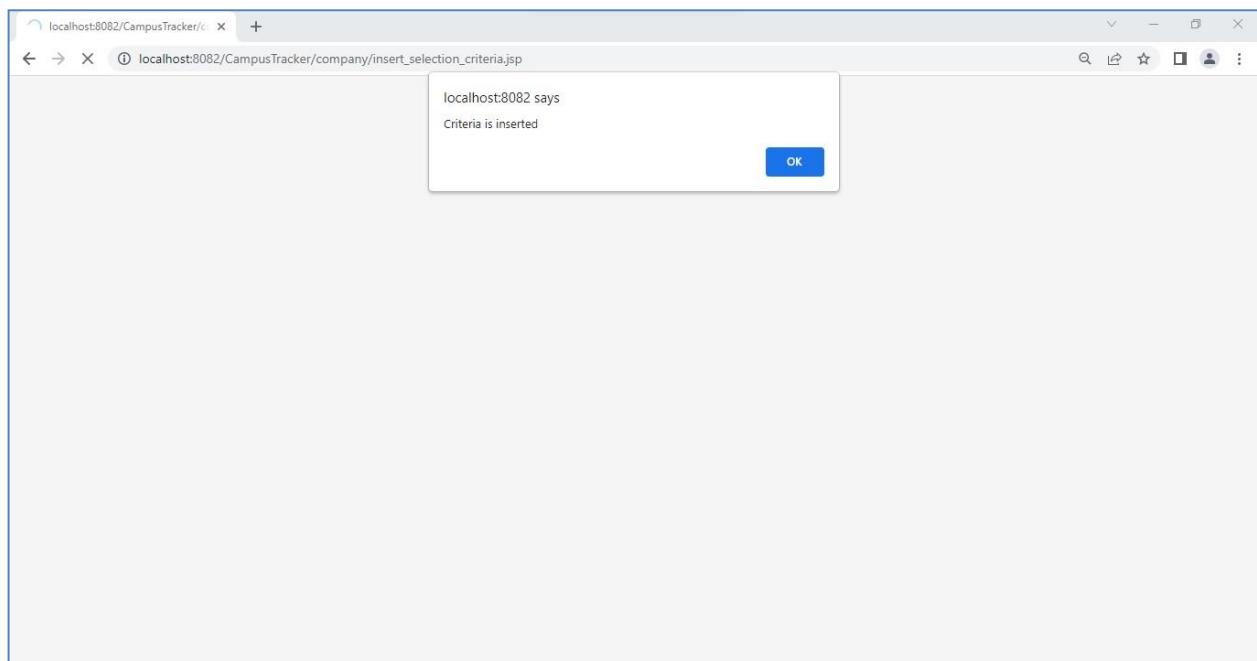
Enter selection Criteria

Company	TCS
SSC	80
HSC	70
Degree	60
Master	50

Submit Reset

Developed by Group no.- 27

Criteria Values inserted Successfully



# CAMPUS TRACKER

## Updated Values in Criteria Information

Admin Home Page

localhost:8082/CampusTracker/company/view\_criteria.jsp

Campus Tracker

View Criteria

Selection Criteria						
Id	Company Name	SSC	HSC	DEGREE	Masters	Delete
1	TCS	60	50	75	70	<a href="#">Delete</a>
2	TCS	80	70	60	50	<a href="#">Delete</a>

No data available in table

Showing 0 to 0 of 0 entries

← Previous Next →

Developed by Group no.- 27

## Criteria Values Deleted Successfully

localhost:8082 says  
Value is deleted...

OK

# CAMPUS TRACKER

## After Deleted Remaining Values

The screenshot shows a web browser window titled "Admin Home Page" with the URL "localhost:8082/CampusTracker/company/view\_criteria.jsp". The page has a blue header with the text "Campus Tracker" and a group of people icon. On the left, there is a sidebar menu with options: Home, College Details, Placement officer, Job Post, Selected Students, criteria, and Change Password. The main content area is titled "View Criteria" and contains a table with one row of data:

Id	Company Name	SSC	HSC	DEGREE	Masters	Delete
1	TCS	60	50	75	70	<a href="#">Delete</a>

Below the table, it says "No data available in table". At the bottom, there are links for "Showing 0 to 0 of 0 entries", "← Previous", and "Next →".

Developed by Group no - 27

## About Page/Contact Page

The screenshot shows a web browser window titled "Project" with the URL "localhost:8082/CampusTracker/". The page has a light blue header with the text "ABOUT".

On the left, there is a sidebar with three circular icons: a white circle with a black outline, a blue circle with a white outline, and a grey circle with a white outline.

In the center, there is a logo for "ASIMBMR" featuring a globe icon and the text "ASIMBMR" in large blue letters, with "Education • Research • Consultancy" below it.

Below the logo, there is a large blue button with the text "IBMR" in white.

On the right, there is a section titled "INSTITUTE OF BUSINESS MANAGEMENT AND RESEARCH" with the following text:  
The mission of IBMR is to maintain & build excellence in the youth, unleash their competencies with creative & analytical potentials in the Management field. IBMR has signed the Work Force Development Program (WDP) with Oracle, IBM.

Below this, there is a list of contact information:

- **Departments:** MCA & MBA
- **Website:** <https://www.asmibmr.edu.in/>
- **Phone:** +91 94220 09207/09/10/12
- **Email:** [asmplacement@asmdu.org](mailto:asmplacement@asmdu.org)
- **Address:** ASM Group of Institutes MIDC Block C  
Chinchwad Pune Maharashtra 411019

# CAMPUS TRACKER

## 8. Validation checks

### Student login validation

The screenshot shows the 'Admin Home Page' at [localhost:8082/CampusTracker/user/change\\_password.jsp](http://localhost:8082/CampusTracker/user/change_password.jsp). The page title is 'Campus Tracker'. A sidebar on the left lists navigation options: Home, My Profile, Resume, Selected students, Job details, college details, Student Marks, placement officer, Selected in criteria, and Change Password. The main content area is titled 'Change Password' and contains three input fields: 'Old password', 'New password', and 'Confirm password', each with a red validation message: '\* This field is required' and '\* Letters only'. Below the fields are 'Submit' and 'Reset' buttons.

### Admin login Validation

The screenshot shows the 'Admin Home Page' at [localhost:8082/CampusTracker/admin/student\\_details.jsp](http://localhost:8082/CampusTracker/admin/student_details.jsp). The page title is 'Campus Tracker'. A sidebar on the left lists navigation options: Home, College Details, Placement officer, Student details, Students Resume, Company Details, and Change Password. The main content area is titled 'Student Details' and contains several input fields with validation messages: 'Student name' and 'Student full name' both require letters only; 'Student DOB' requires a date in dd-mm-yyyy format; 'Student gender' requires either male or female selection; 'Student email id' requires a 10-digit phone number; 'Student contact number' and 'Student register number' both require a value; and 'College id' is a dropdown menu with a validation message. Below the fields are 'Submit' and 'Reset' buttons.

# CAMPUS TRACKER

Admin Home Page    localhost:8082/CampusTracker/admin/college\_details.jsp

## Campus Tracker

College Details

College name	<input type="text"/>	* This field is required * Letters only
Principal	<input type="text"/>	* This field is required * This field is required
Email id	<input type="text"/>	* Enter a 10 digit phone number * Numbers only
Contact no	<input type="text"/>	* This field is required * Numbers only
College code	<input type="text"/>	

Submit   Reset

Developed by - Group No. 27

Admin Home Page    localhost:8082/CampusTracker/admin/placement\_officer.jsp

## Campus Tracker

Placement Officer

Placement officer name	<input type="text"/>	* This field is required * Letters only
Qualification	<input type="text"/>	* This field is required * Invalid email address
Email id	<input type="text"/>	* This field is required * Enter a 10 digit phone number
Contact number	<input type="text"/>	

Submit   Reset

Developed by - Group No. 27

# CAMPUS TRACKER

## Company login validation

Admin Home Page    localhost:8082/CampusTracker/company/selection\_criteria\_form.jsp

### Campus Tracker

Company Details

Enter selection Criteria

Company	TCS	* This field is required * Numbers only * This field is required * Numbers only * This field is required * to choose code * This field is required * Numbers only
SSC		
HSC		
Degree		
Master		

Submit   Reset

Developed by Group no.- 27

Admin Home Page    localhost:8082/CampusTracker/company/job\_details.jsp

### Campus Tracker

Job Details

Job Name	TCS	* This field is required * Letters only
Company Name	TCS	* This field is required * Numbers only * This field is required * Letters only * This field is required * Letters only * This field is required * This field is required * Invalid date, must be in YYYY-MM-DD format
Number of jobs		
Job Type		
Skills		
Experience		
Job last date		

Submit   Reset

# CAMPUS TRACKER

## Placement officer login validation

The screenshot shows the 'Campus Tracker' Admin Home Page. On the left, there's a sidebar with a 'Home' button and several menu items: College details, Campus Details, Student details, Resume, Shortlisted students, Selected students, and Change Password. The main content area is titled 'Campus Details'. It contains several input fields with validation messages: 'Campus Name' (red box: \* This field is required), 'Placement Officer Id' (red box: \* This field is required), 'Company name' (red box: \* This field is required, \* Letters only), 'Skills' (empty), 'Job type' (red box: \* This field is required), 'Experience' (red box: \* This field is required, \* Letters only), and 'Campus State' (empty). The top navigation bar shows the URL 'localhost:8082/CampusTracker/pl\_officer/campus\_details2.jsp' and a user icon labeled 'Officer'.

## Sign Up Page validation

The screenshot shows the 'Student Registration' page. The background features a chalkboard with a pencil holder containing various colored pencils and pens. The form fields include: 'Name \*' (red box: Please fill in this field.), 'Father Name \*' (red box: Please fill in this field.), 'Phone Number \*' (empty), 'Birth Date \*' (MM/DD/YYYY), 'Gender \*' (Gender dropdown), 'Register \*' (empty), 'College \*' (College dropdown), 'Email-Id \*' (empty), and a 'SUBMIT' button. A 'Back TO Login' link is located at the bottom left of the form area.

## 9. Sample Code

### Database Connection Code-

```
package placement;  
import java.sql.*;  
public class dbconnect  
{  
    private Connection con;  
    public Statement stmt;  
    public String getConn()  
    {  
        try  
        {  
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");  
            con=DriverManager.getConnection("Jdbc:Odbc:vanaja");  
            stmt=con.createStatement();  
        }  
        catch(Exception e)  
        {  
            System.out.println(e);  
        }  
        return"";  
    }  
}
```

## CAMPUS TRACKER

---

### Form Code –

```
<jsp:include page="meta_tags.jsp"></jsp:include>

<body>

<jsp:include page="header.jsp"></jsp:include>

<jsp:include page="sidebar.jsp"></jsp:include>

<!--/span-->

<div class="span9" id="content">

    <!-- morris stacked chart -->

    <div class="row-fluid">

        <!-- block -->

        <div class="block">

            <div class="navbar navbar-inner block-header">

                <div class="muted pull-left">student details </div>

            </div>

            <div class="block-content collapse in">

                <div class="span12">

                    <link rel="stylesheet" href="js/validationEngine.jquery.css" type="text/css" media="screen" title="no title" charset="utf-8" />

                    <script src="js/jquery.min.js" type="text/javascript"></script>

                    <script src="js/jquery.validationEngine-en.js" type="text/javascript"></script>

                    <script src="js/jquery.validationEngine.js" type="text/javascript"></script>
```

## CAMPUS TRACKER

---

```
<script>
    $(document).ready(function() {
        $("#formID").validationEngine()
    });
</script>

<form name="form1" id="formID" method="post" action="student_details_insert.jsp">
<div align="center">
    <p class="style1">Student details </p>
    <table width="296" border="1">
        <tr>
            <td width="177">Student name </td>
            <td width="103"><input name="stud_name" type="text" id="stud_name" class="validate[required,custom[onlyLetter]]"/></td>
        </tr>
        <tr>
            <td>Student full name </td>
            <td><input name="stud_fullname" type="text" id="stud_fullname" class="validate[required,custom[onlyLetter]]"/></td>
        </tr>
        <tr>
            <td>Student DOB </td>
            <td><input name="stud_dob" type="text" id="stud_dob" class="validate[required,custom[date]]"/></td>
        </tr>
        <tr>
            <td>Student gender </td>
            <td><input name="gender" type="radio" value="male">
                male
            </td>
        </tr>
    </table>
</div>
</form>
```

## CAMPUS TRACKER

---

```
<input name="gender" type="radio" value="female">  
female</td>  
</tr>  
<tr>  
    <td>Student email id </td>  
    <td><input name="stud_email_id" type="text" id="stud_email_id"  
class="validate[required,custom[Email]]"></td>  
</tr>  
<tr>  
    <td>Student contact number </td>  
    <td><input name="stud_contact_no" type="text" id="stud_contact_no"  
class="validate[required,custom[Mobile]]"></td>  
</tr>  
<tr>  
    <td>Student register number </td>  
    <td><input name="stud_register_no" type="text" id="stud_register_no"  
class="validate[required,custom[onlyNumber]]"></td>  
</tr>  
<tr>  
    <td>Resume upload </td>  
    <td><input name="resume_upload" type="text" id="resume_upload"  
class="validate[required,custom[onlyLetter]]"></td>  
</tr>  
<tr>  
    <td>College id </td>  
    <td><select name="college_id" id="college_id">  
        <option value="SELECT">SELECT</option>
```

## CAMPUS TRACKER

---

```
<%@page import="java.sql.*"%>
<%@page import="placement.dbconnect"%>
<jsp:useBean id="m" class="placement.dbconnect"/>
<jsp:getProperty name="m" property="conn"/>
<%
ResultSet rs=m.stmt.executeQuery("select * from college_details");
while(rs.next())
{
%>
<option value=<%=rs.getInt("college_id")%>><%=rs.getString("college_name")%></option>
</select></td>
<%
}
%>
</tr>
<tr>
<td colspan="2"><input type="submit" name="Submit" value="Submit"> <input type="reset" name="Reset" value="Reset"></td>
</tr>
</table>
</div>
</form>
</div>
<ul class="pager wizard">
```

## CAMPUS TRACKER

---

```
<li class="previous first" style="display:none;"><a href="javascript:void(0);">First</a></li>

<li class="previous"><a href="javascript:void(0);">Previous</a></li>

<li class="next last" style="display:none;"><a href="javascript:void(0);">Last</a></li>

<li class="next"><a href="javascript:void(0);">Next</a></li>

<li class="next finish" style="display:none;"><a href="javascript:;">Finish</a></li>

</ul>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

<!-- /block -->

</div>

</div>

</div>

<hr>

<jsp:include page="footer.jsp"></jsp:include>

<link rel="stylesheet" href="js/validationEngine.jquery.css" type="text/css" media="screen" title="no title" charset="utf-8" />

<script src="js/jquery.validationEngine-en.js" type="text/javascript"></script>

<script src="js/jquery.validationEngine.js" type="text/javascript"></script>

<script>

$(document).ready(function() {

    $("#formID").validationEngine()
```

```
});  
</script>
```

### Value Insert Code-

```
<%@page import="java.sql.*"%>  
<%@page import="placement.dbconnect"%>  
<jsp:useBean id="m" class="placement.dbconnect"/>  
<jsp:getProperty name="m" property="conn"/>  
<%  
String a1,a2,a3,a4,a5,a6,a7,a8,a9;  
a1=request.getParameter("stud_name");  
a2=request.getParameter("stud_fullname");  
a3=request.getParameter("stud_dob");  
a4=request.getParameter("gender");  
a5=request.getParameter("stud_email_id");  
a6=request.getParameter("stud_contact_no");  
a7=request.getParameter("stud_register_no");  
a8=request.getParameter("resume_upload");  
a9=request.getParameter("college_id");  
  
int k=m.stmt.executeUpdate("insert into student_details  
values(null,'" +a1+ "','" +a2+ "','" +a3+ "','" +a4+ "','" +a5+ "','" +a6+ "','" +a7+ "','" +a8+ "','" +a9+ "')");  
%>  
<script>  
alert("values are inserted");  
document.location="student_details.jsp";  
</script>
```

### Resume View Code-

```
<jsp:include page="meta_tags.jsp"></jsp:include><body>

<jsp:include page="header.jsp"></jsp:include>

<jsp:include page="sidebar.jsp"></jsp:include>

<!--/span-->

<div class="span9" id="content">

    <!-- morris stacked chart -->

    <div class="row-fluid">

        <!-- block -->

        <div class="block">

            <div class="navbar navbar-inner block-header">

                <div class="pull-left muted">

                    <div align="center"><strong>Resume</strong></div>

                </div>

            </div>

            <div class="block-content collapse in">

                <div class="span12">

                    <div align="center">

                        <a href="resume.jsp" class="btn btn-primary">Upload Resume</a>

                    </div>

                </div>

            </div>

        </div>

    </div>

</div>
```

## CAMPUS TRACKER

---

```
<tr>
<th>Student id </th>
<th>Student Name </th>
<th>Reg No</th>
<th>Resume</th>
<th>Delete</th>
</tr>

</thead>
<tbody>
<% @page import="java.sql.*"%>
<%@page import="placement.dbconnect"%>
<jsp:useBean id="m" class="placement.dbconnect"/>
<jsp:getProperty name="m" property="conn"/>
<%
String uname=session.getAttribute("uname").toString();
ResultSet rs=m.stmt.executeQuery("select * from student_details where
stud_register_no='"+uname+"'");
while(rs.next())
{
int id=rs.getInt("stud_id");
String resume = rs.getString("resume_upload");
%>
<tr>
<td><%=id%></td>
<td><%=rs.getString("stud_name")%></td>
<td><%=rs.getString("stud_register_no")%></td>
```

## CAMPUS TRACKER

---

```
<!-- <td><a href="resume/<%=rs.getString("resume_upload")%>">Download  
Resume</a></td> -->  
  
<td>  
  
    <button id="download" type="button" class="btn btn-info" data-id="<%="resume%>">  
Download</button>  
  
</td>  
  
<td><a href="resume_delete.jsp?id=<%=id%>">Remove</a></td>  
  
</tr>  
  
<%  
  
}  
  
%>  
  
</tbody>  
  
</table>  
  
</div>  
  
</div>  
  
<div align="center">  
  
<!-- <ul class="pager wizard">  
  
    <li class="previous first" style="display:none;"><a  
href="javascript:void(0);">First</a></li>  
  
    <li class="previous"><a href="javascript:void(0);">Previous</a></li>  
  
    <li class="next last" style="display:none;"><a  
href="javascript:void(0);">Last</a></li>  
  
    <li class="next"><a href="javascript:void(0);">Next</a></li>  
  
    <li class="next finish" style="display:none;"><a  
href="javascript:;"/>Finish</a></li>  
  
</ul> -->  
  
</div>
```

## CAMPUS TRACKER

---

```
</div>

</div>

</div>

</div>

<div align="center">

</div>

<!-- /block -->

</div>

</div>

</div>

</div>

<hr>

<jsp:include page="footer.jsp"></jsp:include>

<script>

$(document).ready(function(){

    $(document).on("click","#download",function(){

        var download = $(this).data('id');

        //alert(download);

        if(download=="No Resume")

        {

            $('#download').html("Please Upload Resume");

        }

        else if(download=="Inactive")

        {


```

## CAMPUS TRACKER

---

```
$($('#download').html("Please Upload Resume");

//console.log(b);

//document.location=("../user/resume/" + download;

}

else if (download !== "") {

var b = window.open('../user/resume/' + download, '_blank');

}

else {

//$('#download').html("File Not found");

alert("Resume not exist");

}

})

})

</script>
```

### View Selected Student-

```
<jsp:include page="meta_tags.jsp"></jsp:include>

<style type="text/css">
</style>

<body>

<jsp:include page="header.jsp"></jsp:include>

<jsp:include page="sidebar.jsp"></jsp:include>

<!--/span-->

<div class="span9" id="content">

    <!-- morris stacked chart -->

    <div class="row-fluid">

        <!-- block -->

        <div class="block">

            <div class="navbar navbar-inner block-header">

                <div class="muted pull-left">view selected </div>

            </div>

            <div class="block-content collapse in">

                <div class="span12">

<% @page import="java.sql.*"%>

<% @page import="placement.dbconnect"%>

<jsp:useBean id="mm" class="placement.dbconnect"/>
```

## CAMPUS TRACKER

---

```
<jsp:getProperty name="mm" property="conn"/>

<link rel="stylesheet" href="js/validationEngine.jquery.css" type="text/css" media="screen" title="no
title" charset="utf-8" />

<script src="js/jquery.min.js" type="text/javascript"></script>
<script src="js/jquery.validationEngine-en.js" type="text/javascript"></script>
    <script src="js/jquery.validationEngine.js" type="text/javascript"></script>

<script>
$(document).ready(function() {
    $("#formID").validationEngine()
});

</script>
<form name="form1" id="formID" method="post" action="view_selected.jsp">
<div align="center">
<p class="style1">&nbsp;</p>
<table width="301" border="0">
    <tr>
        <td>Company Name </td>
        <td><select name="Name" id="Name">
            <option value="SELECT">SELECT</option>
        </select>
    </td>
<%
ResultSet rsz=mm.stmt.executeQuery("select * from criteria");
while(rsz.next())
{

```

## CAMPUS TRACKER

---

```
%>

<option ><%=rsz.getString("Name")%></option>

    public String s1=rsz.getParameter("Name");

<%
}

%>

<p>

    <input type="submit" name="Submit" class="btn btn-primary"
value="Search">

    <a href="check.jsp"> </a>

</p>

</select></td>

</tr>

</form>

<p><strong></strong></p>

</div>

</div>

</div>

<div class="block-content collapse in">

<div class="span12">

<div align="center">

    <table cellpadding="0" cellspacing="0" border="0" class="table table-striped
table-bordered" id="example">
```

## CAMPUS TRACKER

---

```
<thead>
<tr>
<th scope="row">Student_no</th>
<td><strong>Register_no</strong></td>
<td><strong>Student_name</strong></td>
<td><strong>SSC percentage </strong></td>
<td><strong>HSC percentage </strong></td>
<td><strong>Degree percentage </strong></td>
<td><strong>Masters percentage </strong></td>
</tr>
<%@page import="java.sql.*"%>
<%@page import="placement.dbconnect"%>
<jsp:useBean id="m" class="placement.dbconnect"/>
<jsp:getProperty name="m" property="conn"/>
<%
String s1=request.getParameter("Name");
```

```
ResultSet rs=m.stmt.executeQuery("select * from marks m,criteria c where m.sslc_marks>=c.sslc and m.puc_marks>=c.puc and m.degree_marks>=c.degree and c.Name='"+s1+"'");
```

```
while(rs.next())
{
int student_no=rs.getInt("student_no");
%>
<tr>
<th scope="row"><%=student_no%></th>
<td><%=rs.getString("reg_no")%></td>
<td><%=rs.getString("student_name")%></td>
```

## CAMPUS TRACKER

---

```
<td><%=rs.getString("sslc_marks")%></td>
<td><%=rs.getString("puc_marks")%></td>
<td><%=rs.getString("degree_marks")%></td>
<td><%=rs.getString("masters")%></td>
</tr>
<%
    }
%>
</tbody>
        </table>
    </div>
</div>
        <div align="center">
            </div>
        </div>
    </div>
</div>
        <div align="center">
            </div>
        </div>
    </div>
</div>
<hr>
```

### **10. Implementation and Maintenance:**

The goal of the coding phase is to translate the design into code in the given programming language. The coding steps translate the detailed design of the system into programming language. The translation process continues when compiler accepts source code as input and produces machine dependent object code as output. Linking of object files are done to produce the machine code. Internal documentation is another important factor, to facilitate others to understand the code and the logic.

#### **Code review and walk through:**

Both reviews and walk through used to deliver the correct codes. The code review is done as soon as code is ready to be executed, this is to reduce syntax errors and also check the coding standard.

#### **Module Specification:**

The modules specified in the design are implemented using various ".html", "jsp" and "class" files. These files in the source code share the common routines and share the data structure, to establish the hierarchical relationship.

#### **Compilation and Building the executable:**

The source code for the system organized in various files is compiled using the "java" utility provided in the JAVA. The application is made to run in web browser the address as "<http://localhost:8081/staff>" present in ROOT directory of Tomcat Server.

#### **Running the package:**

The following steps are undertaken to execute this application- First start Tomcat Server. • Open any Web Browser like Google Chrome, Mozilla Firefox. Type the following address in address bar of Web browser. <http://localhost:8080/staff> Now the Home page of application is displayed on screen. If new registration is there then register or sign up first and use the application. If already registered then directly logon into the system.

### **Java:**

Java is a computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that code that runs on one platform does not need to be recompiled to run on another. Java applications are typically compiled to byte code (class file) that can run on any Java virtual machine (JVM) regardless of computer architecture. Java is, as of 2014, one of the most popular programming languages in use, particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by James Gosling at Sun Microsystems (which has since merged into Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++, but it has fewer low-level facilities than either of them.

The original and reference implementation Java compilers, virtual machines, and class libraries were developed by Sun from 1991 and first released in 1995. As of May 2007, in compliance with the specifications of the Java Community Process, Sun relicensed most of its Java technologies under the GNU. Others have also developed alternative implementations of these Sun technologies, such as the GNU Compiler for Java (byte code compiler), GNU Class path (standard libraries), and Iced Tea-Web (browser plug in for applets).

### **Advantages of java:**

- Purely Object oriented.
- Platform independent.
- It is dynamic, simple and robust.
- Easy to learn.
- Multithreaded. Secure.
- Wide variety of Application Programmer Interfaces (APIS).
- Excellent networking capability.

### **The java Platform:**

The Java platform is the name given to the computing platform from Oracle that helps users to run and develop Java applications. The platform does not just enable a user to run and develop Java application, but also features a wide variety of tools that can help developers work efficiently with the Java programming language

**The platform consists of two essential software's:**

- The Java Runtime Environment (JRE), which is needed to run Java applications and applets
- The Java Development Kit (JDK), which is needed to develop those Java applications and applets. If you have installed the JDK, you should know that it comes equipped with a JRE as well. So, for all the purposes of this book, you would only require the JDK.

### **Java Components:**

Java has two components those are

1. Java virtual machine (JVM).
2. Java Application Programmers Interface (API).

### **JVM**

A Java virtual machine (JVM) interprets compiled Java binary code (called byte code) for a computer's processor (or "hardware platform") so that it can perform a Java program's instructions. Java was designed to allow application programs to be built that could be run on any platform without having to be rewritten or recompiled by the programmer for each separate platform. A Java virtual machine makes this possible because it is aware of the specific instruction lengths and other particularities of the platform

### **API-**

An application programming interface (API) is a library of functions that Java provides for programmers for common tasks like file transfer, networking, and data structures.

- Java Program
- Java API
- Java Virtual Machine
- Hardware-Based Platform

### **Java in web:**

Java covers the whole application from server to client and back again, it provides many powerful technologies, it can be used to extend the browser, and it provides good security system. HTML: HTML stands for hypertext markup Language. It is very useful to make web pages and very easy to learn. Hypertext Markup file is a text file containing small markup tags. These marks up tags tell the browser how to display a web page. It has two types of extensions one is html and second is htm but both are used for html web pages. For hypertext markup language you can use the simple text editor for example; use notepad for writing your HTML code in the windows. If you are using Mac you can use simple text editor. HTML uses approach of what you see is what you get. You can also use to write tags other software that is FrontPage and Dreamweaver. In HTML character are surrounded by the tags. HTML tags come in pair. The beauty of this language is that it is not case sensitive. Every web page need HTML with it you cannot make the good web pages. And it is the base for every web page and used to display the text in the web pages there are some other latest version of HTML like DHTML which stands for dynamic html and is used to make the web pages more interactive

### **Features of HTML:**

- It is simple to understand and implement.
- HTML constructs are very easy to comprehend, and can be used effectively by anybody.
- HTML syntax is a worldwide standard.
- The methodology used by HTML to markup information is independent of its representation on a particular hardware or software architecture.
- It is not a programming language.
- And it is also not a description language

### **Java Server Pages (JSP):**

Java Server Pages (JSP) lets you separate the dynamic part of your pages from the static HTML. We simply write the regular html in the normal manner, using whatever Web-page-building tools you normally use. We then enclose the code for the c parts in special tags, most of which start with "<%" and end with %. We normally give your file a jsp extension, and typically install it in any ace you could place a normal Web page. Although what you write often looks more a regular html file than a servlet, behind the scenes, the JSP page just gets converted to a normal servlet, with the static html simply being printed to the output stream associated with the servler's service method. This is normally done the first time the page is requested, and developers can simply request the page themselves when first installing it if they want to be sure that the first real user doesn't get a momentary' delay when the JSP page is translated to a servlet and the servlet is compiled and loaded. Many Web servers let you define aliases that so that a URL that appears to reference an html file really points to a servlet or JSP page.

### **Advantages of jsp:**

- Separation of static from dynamic content.
- Write Once Run Anywhere.
- Recommended Web access layer for n-tier architecture.
- Completely leverages the Servlet API.
- Platform independent.
- Reuse of components and tag libraries.
- Encapsulation of functionality.
- They have a better performance and scalability than ordinary CGI scripts, because they are persistent in memory and multi-threaded.
- They have built in support for HTTP sessions, which makes application Programming possible.
- They have full access to Java Technology-Network awareness, threads and Database connectivity-without the limitations of client side application applets.
- They are automatically recompiled when necessary.
- They exist in the ordinary Web server document space, no special URL mapping is required to address them.

### **How jsp works?**

#### **JSP pages exist in 3 forms or versions**

- JSP source code consists of text file with an extension of jsp and contains a mix of HTML template code, Java language statements and JSP directives and actions that describe how to generate a web page to service a particular request.
- Java source code: the jsp container translates the jsp source code into the source code for an equivalent Java Servlet as needed.
- Compiled Java class: Like any other Java class, the generated servlet code is compiled into byte-codes in a .class file ready to be loaded and executed.

### **Java Script:**

#### **What is Java Script?**

- Java Script is embedded into html.
- It is browser dependent.
- JavaScript depends on the web browser to support it. If the browser doesn't support it, JavaScript code will be ignored. Internet Explorer 3.0 and Netscape Navigator 2.0 onwards support JavaScript.
- It is an interpreted language, loosely typed, object based language.
- Java script is not Java

### **Data Validation:**

JavaScript provides the means for basic data validation before it is sent to the server. Whether the values entered are correct or not or whether all the fields in a form are filled out or not can be checked before sending data to web server, if JavaScript is not used then data is sent to web server, and the web server would response with a message that the data sent to it is incorrect or incomplete. Thus JavaScript ensures data validation and also reduces the network traffic.

### **Reasons for Back end:**

As Oracle 8.0 is object oriented and robust RDBMS & a number of tools available to work with it make it quite attractive as a backend.

In a market place populated by computer companies with "proprietary" hardware, "proprietary" operating systems, "proprietary" databases and "proprietary" applications, ORACLE gives business users systems departments' new control over their lives and futures. They are no longer bound to the database product of a single hardware vendor. ORACLE can run on nearly every kind of computer they can own. This is a basic revolution in the workplace and in application development, with consequences that will extend far into the future.

### **Java Data Base Connectivity (JDBC):**

In an enterprise computing which is largely the black art of managing huge databases? People associated with the enterprise need to be able to use and update the data easily, quickly and securely,

The powerful Java Data Base Connectivity (JDBC) suit the java.sql. package realizes java's promise as a serious business programming tools.

## CAMPUS TRACKER

---

Java Data Base Connectivity is a standard SQL database access interface providing uniform access to a wide range of relational databases. It also provides a common base on which higher level tools and interfaces can be built. This comes with an "ODBC Bridge". That bridge is a library, which implements JDBC in terms of ODBC standard API.

There are many types of drivers used in connecting such as Native API Partly Java driver, a net protocol all java driver. The driver used here is JDBC- ODBC Bridge.

JDBC-ODBC bridge plus ODBC driver -This is the crudest possible solution. Applets access data base using a combination of the JDBC-ODBC Bridge and an ODBC driver. This requires both drivers to be installed on the user's computer- A very cumbersome solution for both internet and intranet users.

JDBCTM is a Java Tm API for executing SQL statements. (As a point of interest, JDBC is a trademarked name and is not an acronym; nevertheless, JDBC is often thought of as standing for "Java Database Connectivity".) It consists of a set of classes and interfaces written in the Java programming language . JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API.

Using JDBC, it is easy to send SQL statements to virtually any relational database. In other words, with the JDBC API, it isn't necessary to write one program to access a Sybase database, another program to access an MYSQL

ie, da database, another program to access an Informix database, and so on. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. And, with an application written in the Java programming language, one also doesn't have to worry about writing different applications to run on different platforms. The combination of Java and JDBC lets a programmer write it once and run it anywhere.

Java being robust, secure, easy to use, easy to understand, and automatically downloadable on a network, is an excellent language basis for database applications.

What is needed is a way for Java applications to talk to a variety of different databases. JDBC is the mechanism for doing this. JDBC extends what can be done in Java. For example, with Java and the JDBC API, it is possible to publish a web page containing an applet that uses information obtained from a remote database. Or an enterprise can use JDBC to connect all its employees (even if they are using a conglomeration of Windows, Macintosh, and UNIX machines) to one or more internal databases via an intranet. With more and more programmers using the Java programming language, the need for easy database access from Java is continuing to grow.

MIS managers like the combination of Java and JDBC because it makes disseminating information easy and economical. Businesses can continue to use their installed databases and access information easily even if it is stored on different database management systems. Development time for new applications is short. Installation and version control are greatly simplified. A programmer can write an application or an update once, put it on the server, and everybody has access to the latest version. And for businesses selling information services, Java and JDBC offer a better way of getting out information updates to external customers.

### **JDBC does the following things:**

- Establish a connection with a database
- Send SQL statements
- Process the results.

### **The following code fragment gives a basic example of these three steps:**

```
Class.forName ("sun.jdbc.odbc.JdbcOdbcDriver");  
  
Connection con=DriverManager.getConnection("Jdbc: Odbc:dsnname");  
  
Statement stmt=con.createStatement();
```

### **Connection:**

A connection object represents a connection with a database. A connection session includes the SQL statements that are executed and the results that are returned over the connection. A single application can have one or more connections with a single database, or it can have connections with many different databases.

### **Opening a Connection:**

The standard way to establish a connection with a database is to call the method `DriverManager.getConnection`. This method takes a string containing URL. The Driver Manager class, referred to as the JDBC management layer, attempts to locate a driver than can connect to the database represented Driver classes, and when the method `get Connection` is called, it checks with each driver in the list until it finds one that can connect uses this URL to actually establish the connection.

### **Sending Statement:**

Once a connection is established, it is used to pass SQL statements to its underlying database. JDBC does not put any restrictions on the kinds of SQL statements that can be sent; this provides a great deal of flexibility, allowing the use of database-specific statements or even Non-SQL statements. It requires, however, that the user be responsible for making sure that the underlying database can process the SQL statements being sent and suffer the consequences if it cannot.

### **Driver Manager:**

The Driver Manager class is the management layer of JDBC, working between the user and the drivers. It keeps track of the drivers that are available and handles establishing a connection between a database and the appropriate driver. In addition, the driver manager class attends to things like driver login time limits and the printing of log and tracing messages. The only method in this class that a general programmer needs to use directly is `DriverManager.getConnection`. As its name implies, this method establishes a connection to a database.

### **Why we need JDBC?**

- ODBC is not appropriate for direct use from Java because it uses C interfaces.
- ODBC is hard to learn. It mixes simple and advanced features together, and it has complex options even for simple queries.
- A Java API like JDBC is needed in order to enable a "Pure Java" solution.
- When ODBC is used, the ODBC driver manager and drivers must be manually installed on every client machine.

### **SESSION:**

This is the `HttpSession` object associated with the request. Recall that sessions are created automatically, so this variable is bound even if there was no incoming session reference. The one exception is if you use the `session` attribute of the page directive to turn sessions off, in which case attempts to reference the `session` variable cause errors at the time the JSP page is translated into a servlet.

### **Apache Tomcat:**

Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and Java Server Pages technologies. The Java Servlet and Java Server Pages specifications are developed by Sun under the Java Community Process. Tomcat is a web server that supports servlets and JSPs. Tomcat is available for commercial use under the ASF license from the Apache Tomcat comes with the Jasper compiler that compiles JSPs into servlets. website in both binary and source versions.

Apache Tomcat is developed in an open and participatory environment and released under the Apache Software License. Apache Tomcat powers numerous large scale, mission-critical web applications across a diverse range of industries and organizations. Different versions of Apache Tomcat are available for different versions of the Servlet and JSP specifications. The Tomcat servlet engine is often used in combination with an Apache web server or other web servers. Tomcat can also function as an independent web server. Earlier in its development, the perception existed that standalone

## CAMPUS TRACKER

---

Tomcat was only suitable for development environments and other environments with minimal requirements for speed and transaction handling. However, that perception no longer exists; Tomcat is increasingly used as a standalone web server in high-traffic, high availability environments. Since its developers wrote Tomcat in Java, it runs on any operating system that has a JVM.

### **Usage:**

Java Web Applications, Java Mobile Applications using J2ME

### **Access:**

Create a Java Web Application or Mobile Application using Net Beans Build and run project, this will automatically launch the Apache Tomcat as default.

### **Viewing Web Applications:**

<http://localhost: 8080/WebpageName.jsp>

### **Directory Structure:**

The typical and default directory hierarchy of a Tomcat installation comprises the following:

- bin – startup , shutdown and other scripts and executable.
- common - common classes that Catalina and web applications can use.
- conf - XML files and related DTDS to configure Tomcat.
- logs - Catalina and application logs.
- server - classes used only by Catalina.
- shared - classes shared by all web applications.
- webapps - directory containing the web applications.
- work - temporary storage for files and directories.
- 

### **A web application is basically a web site that:**

- "Knows who you are"--it doesn't just give you static pages, it interacts with you.
- Can permanently change data (such as in a database).
- A web application can consist of multiple pieces.
- Static web pages (possibly containing forms).
- Servlets.
- JSP.

Tomcat organizes all these parts into a single directory structure for each web application.

### The flow that takes place is:

- The user submits an HTML form,
- Tomcat finds the servlet based on the URL and the deployment descriptor, (web.xml) and passes the request to the servlet.
- The servlet computes a response.
- The servlet writes an HTML page containing the response.
- The servlet forwards the response to the JSP.
- The JSP embeds the response in an HTML page
- Tomcat returns the HTML page to the user.

### Status:

Tomcat is available at the Jakarta binary downloads page. The Tomcat server is a Java based Web Application container that was created to run Servlets and Java Server Pages (JSP) in Web applications. As part of Apache's open source Jakarta project, it has nearly become the industry accepted standard reference. Implementation for both the Servlets and JSP API. Written by expert Servlets and JSP software architect and author James Goodwill, this column will feature introductory Web application development issues, Tomcat installation and configuration, deploying Web applications onto Tomcat, Struts and much more. Apache Tomcat (formerly under the Apache Jakarta Project; Tomcat is now a top level project) is a web container developed at the Apache Software Foundation. Tomcat implements the servlet and the Java Server Pages (JSP) specifications from Sun Microsystems, providing an environment for Java code to run in cooperation with a web server. It adds tools for configuration and management but can also be configured by editing configuration files that are normally XML formatted. Because

Tomcat includes its own HTTP server internally, it is also considered a standalone web server, Tomcat is a web server that supports servlets and JSPs. Tomcat comes with the Jasper compiler that compiles JSPs into servlets. The Tomcat servlet engine is often used in combination with an Apache webserver or other web servers. Tomcat can also function as an independent web server

Earlier in its development, the perception existed that standalone Tomcat was only suitable for development environments and other environments with minimal requirements for speed and transaction handling. However, that perception no longer exists;

Tomcat is increasingly used as a standalone web server in high-traffic, high availability environments. Since its developers wrote Tomcat in Java, it runs on any operating system that has a JVM.

## CAMPUS TRACKER

---

### **Beans:**

Its used to achieve reusability in java it is introduced to overcome the drawbacks of the traditional inheritance and object relations in java bean. The reusability is achieved using bean class. Bean is a component equivalent to active component of visual basic. To create beans we must follow design pattern rules,

- The class must be placed inside the package
- Class must be defined as public
- Variables in the class are defined as private to avoid direct Accessibility of variables.
- Variables must be defined in lower case

A bean class can have public variable also, for every private variable there must be Set and Get methods. Set takes parameter which is used to assign the values for the variable. Get is used to retrieve the values. Every bean class should have implicit constructor (default constructor).

### **Functional Requirement:**

#### **Admin:**

Admin has given permission to add student and to provide login for student. He can also view the details of other colleges and also has given permission to view the company and campus details.

#### **User:**

Student is a one of the main module in our project here we have provided the menus options like view company Upload resumes ,view selected in criteria, upload marks, view selected students and change password. In view company students can get all the details of the company, name of the company, job titles, on which date the company is visiting our college etc.

#### **Placement officer:**

Placement officer can login his account using password and username. He can see the college details and also see the campus details and placement officer can add and see the all student details and also download the resumes of students and see the selected in the campus.

#### **Company:**

Login is provided for company when they will fill their details like name of the company, password and email. When a company is planned to conduct campus interview for the colleges they add the job details like requirements and conditions by entering into our college web site. And also add the company criteria. He can also view the details of other colleges and also has given permission to view the company and campus details. And also select the student who has selected in campus.

## CAMPUS TRACKER

---

### **System Maintenance:**

The term "software maintenance" is used to describe the software engineering activities that occur following delivery of a software product to the customer. The maintenance phase of the software life cycle is the time period in which a software product performs useful work. Maintenance activities involve making enhancement to software products, adapting products to new environments and correcting problems. Software product enhancement may involve providing new functional capabilities, improving user display and modes of interaction, and upgrading external documents. Adaptation of software to a new environment may involve moving the software to a different machine. Problem correction involves modification and revalidation of software to correct errors. The enhancement of this project can be accomplished easily. That is, any new functional capabilities can be added to the project by simply including the new module in the homepage and giving a hyperlink to that module. Adaptation of this project to a new environment is also performed easily.

Even with the best quality assurance activities, it is likely that the customer will uncover defects in the software. Corrective maintenance changes the software to correct defects

**11. Future scope of the Mini Project:**

- We can directly prepare a resume and upload it into the system instead of uploading separately.
- We can upload placement papers so that students can solve those and get to know the scores also.
- We can add a message/ E-mail system so that a placement officer and the company can communicate over the mail.

**12. Bibliography/References/Glossary:**

- Herbert Schildt "Java 2-The Complete Reference"
- Phill Hann "JSP-The Complete Reference"
- "Learn Java in 21 days "by SAM's publications
- Software Engineering by Roger s. Pressman
- J2EE Bible By James McGovern

**Websites:**

- <http://Java.sun.com/j2ee/fag/html>
- <http://Java.sun.com/products/jdbc/reference/faqs/index.html>
- <http://www.apt.jhu.edu/~hall/java/servlet-Tutorial>